ON THE STABILITY OF POISEUILLE FLOW

William Francis Harrison

NAVAL POSTGRADUATE SCHOOL
Monterey, California

THESIS

ON THE STABILITY OF POISEUILLE FLOW

by

William Francis Harrison

September 1975

Thesis Advisor: T. H. Gawain

Approved for public release; distribution unlimited.

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE

READ INSTRUCTIONS BEFORE COMPLETING FORM

1. REPORT NUMBER	2. GOVT ACCESSION NO.
3. RECIPIENT'S CATALOG NUMBER	
4. TITLE (and Subtitle) On the Stability of Poiseuille Flow	5. TYPE OF REPORT & PERIOD COVERED Engineer's Thesis; September 1975
	6. PERFORMING ORG. REPORT NUMBER
7. AUTHOR(s) William Francis Harrison	8. CONTRACT OR GRANT NUMBER(s)
9. PERFORMING ORGANIZATION NAME AND ADDRESS Naval Postgraduate School Monterey, California 93940	10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS
11. CONTROLLING OFFICE NAME AND ADDRESS Naval Postgraduate School Monterey, California 93940	12. REPORT DATE September 1975
	13. NUMBER OF PAGES 163
14. MONITORING AGENCY NAME & ADDRESS (if different from Controlling Office) Naval Postgraduate School Monterey, California 93940	15. SECURITY CLASS. (of this report) Unclassified
	15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)

Approved for public release; distribution unlimited.

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

18. SUPPLEMENTARY NOTES

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

Poiseuille flow
Stability of fluid flow

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

The three-dimensional linearized vorticity transport equations for plane and pipe Poiseuille flow were studied using a highly generalized complex exponential form of solution in both space and time. The stability of these flows was examined using frames of reference which move with the fluid particles.

Numerical results for plane Poiseuille flow show that the critical Reynolds number is lowered by the introduction of streamwise spatial decay. This result provides a new basis for improving the agreement between theory and experiment. Numerical results for pipe flow were not obtained due to a probable error in some detail of the analysis or numerical method.

On the Stability of
Poiseuille Flow

by

William Francis Harrison
Lieutenant, United States Navy
B.S., Rensselaer Polytechnic Institute, 1966

Submitted in partial fulfillment of the

requirements for the degree of

AERONAUTICAL ENGINEER

from the

NAVAL POSTGRADUATE SCHOOL
September 1975

ABSTRACT

The three-dimensional linearized vorticity transport equations for plane and pipe Poiseuille flow were studied using a highly generalized complex exponential form of solution in both space and time. The stability of these flows was examined using frames of reference which move with the fluid particles.

Numerical results for plane Poiseuille flow show that the critical Reynolds number is lowered by the introduction of streamwise spatial decay. This result provides a new basis for improving the agreement between theory and experiment. Numerical results for pipe flow were not obtained due to a probable error in some detail of the analysis or numerical method.

TABLE OF CONTENTS

LIST OF FIGURES

TABLE OF SYMBOLS

All quantities below except those subscripted with d are in dimensionless form. The nondimensionalization is described in Appendix A.

$D, D^2, \ldots$	The partial derivatives with respect to y in cartesian coordinates or with respect to r in cylindrical coordinates.
e	Base of natural logarithms. Exponentiation is also denoted by exp( )
$\bar{e}_x, \bar{e}_r, \bar{e}_\theta$	Unit vectors along the x, r, and θ axis in cylindrical coordinates.
f, g, h	Components of the velocity vector potential defined in equations (2-9a) and (E-1).
i	$+\sqrt{-1}$, the imaginary unit. Also used as index in finite differencing mesh in Section III.
$\bar{i}, \bar{j}, \bar{k}$	Unit vectors along the x, y, and z axis in cartesian coordinates.
L	Reference length. Radius of pipe in cylindrical coordinates and semiheight of channel in cartesian coordinates defined in Appendix A.
n	The number of interior points in the finite differencing mesh of Section III. Also used as imaginary part of β in pipe flow. That is, $\beta = \beta_R + in$.

p Pressure.

Re Reynolds number based on mean velocity and channel semiheight or pipe radius.

t Time.

T, t Shorthand notation for commonly occuring groups of symbols defined in equations (D-54), (E-34) and (E-48).

u, v, w Components of the complex perturbation velocity defined in equation (2-9b).

u_n, v_n, w_n Components of arbitrary vector function $\bar{V}_n$.

u'_n, v'_n, w'_n Components of $\bar{V}'_n$.

$\bar{v}$ Velocity vector of the perturbation flow defined in Appendix A.

$\bar{V}$ Velocity vector of unperturbed laminar flow defined in equations (2-3) and (2-5).

V_{avg} Mean velocity defined in Appendix A.

$\bar{V}_n$ Arbitrary vector function in cylindrical coordinates with angular periodicity n. Used in Appendix G.

$\bar{V}'_n$ Vector function $\bar{V}_n$ rotated ϕ about the origin.

$\bar{W}$ Complex vector potential of perturbation

velocity defined by equation (2-7).

x, r, θ — Cylindrical coordinates defined in Figure 1-2.

x, y, z — Cartesian coordinates defined in Figure 1-1.

$\bar{V}_d, p_d, t_d$ — Dimensional variables of velocity, pressure and time used in Equation (1-1).

α — $\alpha_R + i\alpha_I$ Complex wave number of the perturbation in the x direction.

β — $\beta_R + i\beta_I$ Complex wave number of the perturbation in the z or θ direction.

γ — $\gamma_R + i\gamma_I$ Complex frequency of the perturbation.

$\bar{\Gamma}$ — The vorticity transport equation (D-55) expressed in abbreviated notation as defined by equations (B-1) and (B-11).

$\Gamma_x, \Gamma_y, \Gamma_z$ — The components of $\bar{\Gamma}$ in cartesian coordinates defined in equation (B-1).

$\Gamma_x, \Gamma_r, \Gamma_\theta$ — The components of $\bar{\Gamma}$ in cylindrical coordinates defined in equation (B-11).

ν — Kinematic viscosity.

ξ, η, ζ — Components of the complex perturbation vorticity defined in equation (2-9c).

ρ	Density in equation (A-1).
Φ	Angle of rotation of arbitrary point about the origin in cylindrical coordinates as in figure G-1.
$\bar{\omega}$	Vorticity vector of the perturbation flow defined in equation (A-8).
$\bar{\Omega}$	Vorticity vector of unperturbed laminar flow defined in equations (2-4) and (2-6).
∇	Linear vector operator (nabla). Used for gradient (∇), divergence ($\nabla \cdot$) and curl ($\nabla \times$).
$\times$	Sign of vector cross multiplication.
[ ]	Brackets enclosing a matrix.
{ }	Brackets enclosing a column vector.

I. BACKGROUND

The problem of solving the Navier-Stokes equation for the onset of instabilities in laminar flow for simple geometries has been studied for many years. The problem is reduced to its simplest form by linearizing the vorticity transport equation, assuming sinusoidal perturbations in space and complex exponential perturbations in time, and reducing the equations to their two-dimensional form. The solutions to this simplified problem are too stable to agree with experimentally obtained values of the critical Reynolds number.

To compare Reynolds numbers from dissimilar flows, the hydraulic diameter and mean flow are generally used as the characteristic length and velocity. Using these criteria, the critical Reynolds number for a pipe is 2600 [Schlichting 1968] and for a rectangular channel with a width to height ratio of 8:1 is also 2600 [Kao and Park 1969]. The analysis in this paper uses the channel semiheight and the pipe radius for analytic convenience (with the mean velocity). Based on these characteristic lengths the experimentally obtained critical Reynolds numbers become 1300 for a pipe and 730 for the 8:1 channel. According to Kao and Park the critical Reynolds number for plane Poiseuille flow will be greater than that for a channel of finite width.

The possible destabilizing effect of three-dimensional flow in the case of plane Poiseuille flow was ruled out by Squire's theorem [Squire 1933] which proved that three-dimensional perturbations become more unstable as they are rotated into the plane of the two-dimensional flow. Generalizing the problem by allowing exponential growth of the perturbations in space as well as time greatly adds to the complexity, and the solution of the general problem has not been previously approached.

Much work has been done for the case of pipe flow. Flows with various angular wave numbers, and with sinusoidal spatial perturbations were shown to be stable [Salwen and Grosch 1972], perturbations with exponential growth in space, but with strictly sinusoidal time variation were studied and found stable [Garg and Rouleau 1971] and the combination of exponential growth in space and in time was studied using power series analysis [Gill 1973] and all were found to be stable.

Because of the general agreement that pipe flow is stable to infinitesimal disturbances, two other approaches to the problem of accounting for the critical Reynolds number have been used. First, it has been assumed that perturbations which are stable while they are infinitesimal, become unstable when they are finite. Studies have concluded that finite disturbances are unstable for pipe flow [Davy and Drazin 1968]. Similar theoretical examination of plane flow has reached the same conclusion, that is, finite perturbations are less stable than infinitesimal perturbations [McIntire and Lin 1971]. Second, it has been postulated that the origin of the critical instabilities occurs in the entrance region of the pipe (or channel) and both theoretical and experimental studies have been done to demonstrate this [Huang and Chen 1973] [Leite 1956].

These two explanations for the observed instability of Poiseuille flows have been postulated as a result of the inability of the linear theory to account for the experimental facts. Unfortunately, a totally general solution to the linear problem has never been accomplished. In this paper such a solution is presented. Perturbations which have a fully complex exponential form and which are fully three-dimensional are introduced and the resulting equations are solved using finite-difference methods.

Results for channel flow indicate that by varying the real part of the spatial wave number, the corresponding critical Reynolds number can be lowered. This is promising because it might help resolve discrepancies between theory and experiment. Further study is necessary to clarify these results.

II. THEORY

A. BASIC EQUATIONS

The unperturbed laminar flow of an incompressible fluid with constant viscosity is governed by the vorticity transport and continuity equations. These involve the velocity and vorticity vector functions $\bar{V}$ and $\bar{\Omega}$ which are well known for the two simple geometries being considered (see equations (2-3) through (2-6)). If the flow is slightly perturbed and the perturbation velocity and vorticity are denoted by $\bar{v}$ and $\bar{\omega}$ respectively, the perturbation continuity equation and the linearized vorticity transport equation are easily derived (see Appendix A). These equations, in their nondimensional form, may be written

$$\nabla \bullet \bar{v} = 0 \tag{2-1}$$

and

$$\frac{1}{Re} \nabla^2 \bar{\omega} + \bar{\omega} \bullet \nabla \bar{V} - \bar{V} \bullet \nabla \bar{\omega} + \bar{v} \bullet \nabla \bar{\Omega}$$

$$- \bar{\Omega} \bullet \nabla \bar{v} - \partial \bar{\omega} / \partial t = 0 \tag{2-2}$$

where $\bar{\omega} = \nabla \times \bar{v}$. For the two cases under consideration, plane Poiseuille flow and pipe flow, the coordinate axis orientation is given in figures 2-1 and 2-2 with the velocity and vorticity distributions $\bar{V}$ and $\bar{\Omega}$.

For Plane Poiseuille Flow

$$\bar{V} = \bar{i}\frac{3}{2}(1-y^2) = \bar{i}U(y) \tag{2-3}$$

$$\bar{\Omega}(y) = \nabla\times\bar{V}(y) = \bar{k}3y \tag{2-4}$$

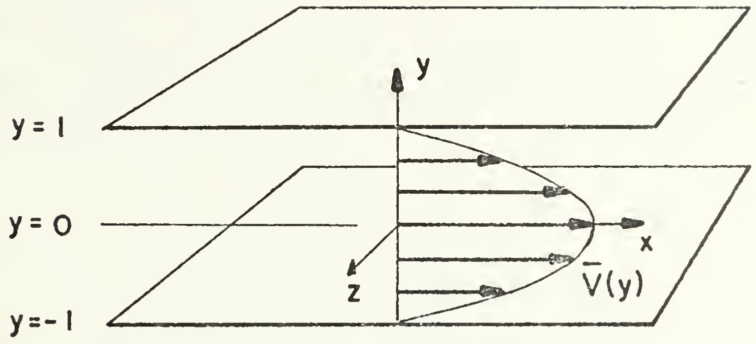

Figure 2-1 Plane Poiseuille Flow

For Pipe Flow

$$\bar{V}(r) = \bar{e}_x U(r) = \bar{e}_x 2(1-r^2) \tag{2-5}$$

$$\bar{\Omega}(r) = \nabla\times\bar{V}(r) = \bar{e}_\theta 4r \tag{2-6}$$

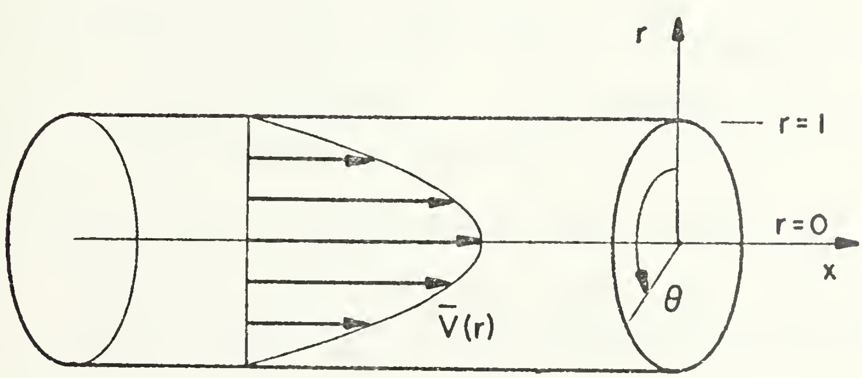

Figure 2-2 Cylindrical Poiseuille Flow

The vorticity transport equation in three dimensions is actually three separate equations. These three equations are not independent (as shown in Appendix B) and represent only two fully independent conditions. The continuity equation is a scalar equation and the relation between vorticity and velocity is a vector equation. Thus there are six independent equations in six unknowns. The unknowns are the three components of the perturbation velocity and the three components of the perturbation vorticity.

These can be reduced to three equations in three unknowns by replacing $\bar{\omega}$ in equations (2-1) and (2-2) by $\nabla \times \bar{v}$. A further simplification can be made by introducing the velocity vector potential, $\bar{W}$, where

$$\bar{v} = \nabla \times \bar{W} \ . \tag{2-7}$$

The continuity equation is satisfied identically because

$$\nabla \cdot \bar{v} = \nabla \cdot (\nabla \times \bar{W}) \equiv 0 \tag{2-8}$$

by a well known vector identity. If $\bar{v}$ is now replaced by $\nabla \times \bar{W}$, the result is two equations (the two independent conditions of the linearized vector vorticity transport equation) in three unknowns. The unknowns are the three components of the velocity vector potential. From this result it may be deduced that the components of $\bar{W}$ are redundant. This is indeed the case and, as shown in Appendix C, one of them may be set arbitrarily to zero.

A useful way to take advantage of the linearity of equation (2-2) is by seeking solutions which are complex. Both the real and imaginary parts of any solution obtained

will, by themselves, be solutions. By choosing solutions of an exponential form, as in equations (2-9), the system of partial differential equations is reduced to ordinary differential equations. The desired form for solutions in cartesian coordinates is

$$\bar{W}(x,y,z,t) = [\bar{i}f(y) + \bar{j}g(y) + \bar{k}h(y)]e^{X} \qquad (2\text{-}9a)$$

$$\bar{v}(x,y,z,t) = [\bar{i}u(y) + \bar{j}v(y) + \bar{k}w(y)]e^{X} \qquad (2\text{-}9b)$$

$$\bar{\omega}(x,y,z,t) = [\bar{i}\xi(y) + \bar{j}\eta(y) + \zeta\bar{k}(y)]e^{X} \qquad (2\text{-}9c)$$

where

$$X \equiv \alpha x + \beta z + \gamma t \ . \qquad (2\text{-}10)$$

For the cylindrical case, the same equations, (2-9) and (2-10), apply with x,y,z replaced by x,r,θ. The wave numbers α, β, and γ are, in general, complex. The functions f, g, and h are defined by equation (2-9a). They are the part of the components of $\bar{W}$ which contain the y (or r) dependence and, in general, are complex. Similarly, u, v and w are part of the components of $\bar{u}$. And ξ, η and ζ are part of each component of $\bar{\omega}$, all complex functions of y (or r).

B. CARTESIAN COORDINATES

The plane flow case for cartesian coordinates is considered first. Using equations (2-9) to substitute into equation (A-13), as described in Appendix D, results in an equation in terms of $\bar{W}$.

$$-\frac{1}{Re}\nabla x\nabla x\nabla x\nabla x\bar{W} + \nabla x\bar{V}x\nabla x\nabla x\bar{W} - \nabla x\bar{\Omega}x\nabla x\bar{W}$$

$$- \nabla x\nabla x\frac{\partial\bar{W}}{\partial t} = 0 \qquad (D-1)$$

Appendix D then develops the form of the coefficients of equation (D-1) as matrices. The final equation is

$$[M_4]\begin{Bmatrix}D^4f\\D^4g\\D^4h\end{Bmatrix} + [M_3]\begin{Bmatrix}D^3f\\D^3g\\D^3h\end{Bmatrix} + ([M_2]+\gamma[N_2])\begin{Bmatrix}D^2g\\D^2g\\D^2h\end{Bmatrix}$$

$$+ ([M_1]+\gamma[N_1])\begin{Bmatrix}Df\\Dg\\Dh\end{Bmatrix} + ([M_0]+\gamma[N_0])\begin{Bmatrix}f\\g\\h\end{Bmatrix} = 0 \qquad (D-55)$$

Where D, D^2, ... are the differential operators with respect to y. The matrix coefficients are given in equations (D-46) through (D-54). This vector equation may be expressed in abbreviated form by

$$\bar{\Gamma} = \begin{Bmatrix}\Gamma_x\\ \Gamma_y\\ \Gamma_z\end{Bmatrix} = 0 \qquad (B-1)$$

as discussed in Appendix B. Equation (B-1) is actually three separate equations

$$\Gamma_x = 0 \qquad (B-2a)$$

$$\Gamma_y = 0 \qquad (B-2b)$$

$$\Gamma_z = 0 \qquad (B-2c)$$

Examination of the matrix coefficients of equation (D-55) shows that the highest order derivatives of the

components f, g, and h, of $\bar{W}$ in each of the three equations is as follows

equation	highest order of f	highest order of g	highest order of h
Γ_x	4	3	2
Γ_y	3	2	3
Γ_z	2	3	4

TABLE 2-1

As demonstrated in Appendix C, one of the components of $\bar{W}$ may be set identically to zero, but, the choice of $h(y)=0$ leads to a degeneracy in the equations for the classical case of $\beta=0$. Since it is desirable to have the three-dimensional case reducible to the plane flow, two-dimensional case, it is stipulated that either f or g be set to zero. Appendix F develops the form of the boundary conditions for the three cases of setting f, g, or h to zero. For $g(y)=0$, the boundary condition

$$f(\pm 1) = h(\pm 1) \qquad \text{(F-6a)}$$

is more complicated than for the case of $f(y)=0$. It is therefore desirable to set $f(y)$ to zero. For $f(y)=0$, the boundary conditions are

$$g(\pm 1) = 0 \qquad \text{(F-4a)}$$

$$h(\pm 1) = 0 \qquad \text{(F-4b)}$$

$$Dh(\pm 1) = 0 \quad \text{(F-4c)}$$

From table 2-1, derivatives of $f(y)$ occur to the fourth order. Equations (F-4b) and (F-4c) give four boundary conditions for $f(y)$. Table 2-1 also shows that $g(y)$ occurs to the third order in equations Γ_x and Γ_z, while there are only two boundary conditions in equation (F-4a). It is necessary to reduce the order of g in Γ_x and Γ_z. This can be done by taking a linear combination of Γ_x and Γ_z. Examination of the matrix of equation (D-47) shows that the coefficient of D^3g in equation Γ_x is $\frac{\alpha}{\overline{Re}}$. In equation Γ_z the coefficient of D^3g is $\frac{\beta}{\overline{Re}}$. Thus, the following combination of equations Γ_x and Γ_z will eliminate the terms in D^3g.

$$-\beta\Gamma_x + \alpha\Gamma_z = 0 \quad \text{(2-11)}$$

Appendix B shows that this linear combination of equations Γ_x and Γ_z, and the condition

$$\Gamma_y = 0 \quad \text{(2-12)}$$

are sufficient to satisfy the vorticity transport equation written as in equation (B-1), under the condition that

$$\alpha^2 + \beta^2 \neq 0 \ . \quad \text{(B-9)}$$

The resulting equations to be solved are (2-11) and (2-12) in the unknowns $g(y)$ and $h(y)$. Performing the actual

calculations indicated by equation (2-11) reveals an interesting and extremely convenient fact. Not only the third order derivatives of g disappear from equation (2-11), but g and all its derivatives cancel out of the equation. Setting $f(y)$ to zero, equation (2-11) only contains $h(y)$. Thus, the equations are uncoupled and (2-11) may be solved for $h(y)$ alone. With $h(y)$ known, equation (2-12) may then be solved for $g(y)$. Writing out equation (2-11) yields

$$-\frac{\alpha}{Re}D^4h + \left(\alpha T - \frac{\alpha}{Re}(\alpha^2+\beta^2)\right)D^2h + (3\alpha^2 + \alpha(\alpha^2+\beta^2)T)h$$
$$+ \gamma(\alpha D^2h + \alpha(\alpha^2+\beta^2)h) = 0 \qquad (2\text{-}13)$$

where

$$T \equiv \alpha U - \frac{1}{Re}(\alpha^2+\beta^2) = \frac{3}{2}\alpha(1-y^2) - \frac{1}{Re}(\alpha^2+\beta^2) \qquad (D\text{-}54)$$

The boundary conditions for equation (2-13) are

$$h(\pm 1) = 0 \qquad (F\text{-}4b)$$

$$Dh(\pm 1) = 0 \ . \qquad (F\text{-}4c)$$

This homogeneous differential equation is solvable as an eigenvalue problem. This technique is discussed in the section on numerical methods. The associated problem is to solve equation (2-12) for $g(y)$. The coefficients of (2-12) may be obtained directly from equations (D-48) through (D-54). This inhomogeneous equation may be solved by various integration techniques to obtain the associated function of $g(y)$ for each of the eigenvalues obtained from the solution of equation (2-13). Solving equation (2-12) would be of interest if the functional shape of the perturbation velocities and vorticities were desired, but the problem was not dealt with in this paper.

C. CYLINDRICAL COORDINATES

The equations for pipe flow, in cylindrical coordinates, are develcped in Appendix E. Equation (D-1) is still valid and in abbreviated form may be expressed by

$$\bar{\Gamma} = \left\{ \begin{matrix} \Gamma_x \\ \Gamma_r \\ \Gamma_\theta \end{matrix} \right\} = 0 \qquad \text{(B-11)}$$

as discussed in Appendix B. This vector equation is actually three separate equations.

$$\Gamma_x = 0 \qquad \text{(B-12a)}$$

$$\Gamma_r = 0 \qquad \text{(B-12b)}$$

$$\Gamma_\theta = 0 \qquad \text{(B-12c)}$$

The final matrix form of equation (B-11) is given in equation (E-55) with the coefficient of equations (E-40) through (E-48). Examination of these coefficients shows that the highest crder of derivatives of the components f, g, and h, of $\bar{W}$ in each of the three equations is as follows

equation	highest order of f	highest order of g	highest order of h
Γ_x	4	3	2
Γ_r	3	2	3
Γ_θ	2	3	4

TABLE 2-2

As in the cartesian case, the order of g(r) in equations Γ_x and Γ_θ is too high for the boundary conditions. To reduce the order of derivatives of g(r) in Γ_x and Γ_θ a linear combination is taken. Checking the coefficients of D^3g in both equations it is found that the combination

$$-\frac{\beta}{r}\Gamma_x + \alpha\Gamma_\theta = 0 \tag{2-14}$$

eliminates the third order derivative of g(r). This combination of equations does not, in general, uncouple the problem by eliminating one of the components of the velocity vector potential from the equation. The resulting matrix equation, with all three components of $\bar{W}$ (one of them may later be set to zero) can be represented in the same form as equation (D-55).

$$[M_4]\begin{Bmatrix} D^4f \\ D^4g \\ D^4h \end{Bmatrix} + [M_3]\begin{Bmatrix} D^3f \\ D^3g \\ D^3h \end{Bmatrix} + ([M_2]+\gamma[N_2])\begin{Bmatrix} D^2g \\ D^2g \\ D^2h \end{Bmatrix}$$

$$+ ([M_1]+\gamma[N_1])\begin{Bmatrix} Df \\ Dg \\ Dh \end{Bmatrix} + ([M_0]+\gamma[N_0])\begin{Bmatrix} f \\ g \\ h \end{Bmatrix} = 0 \tag{D-55}$$

but now the matrix coefficients are 2X3 matrices, where the top row is the combination of equations given in (2-14) and the bottom row is equation Γ_r. The coefficients are as follows.

$$[M_4] = \begin{bmatrix} \frac{\beta}{r\mathrm{Re}} & 0 & -\frac{\alpha}{\mathrm{Re}} \\ 0 & 0 & 0 \end{bmatrix} \tag{2-15}$$

$$[M_3] = \begin{bmatrix} \frac{2\beta}{r^2\mathrm{Re}} & 0 & -\frac{2\alpha}{r\mathrm{Re}} \\ \frac{\alpha}{\mathrm{Re}} & 0 & \frac{\beta}{r\mathrm{Re}} \end{bmatrix} \tag{2-16}$$

$$[M_2] = \begin{bmatrix} \frac{\beta t}{r\mathrm{Re}} - \frac{\beta}{r^3\mathrm{Re}} - \frac{\beta T}{r} & -\frac{4\alpha\beta}{r^2\mathrm{Re}} & \alpha T + \frac{3\alpha}{r^2\mathrm{Re}} - \frac{\alpha t}{\mathrm{Re}} \\ \frac{\alpha}{r\mathrm{Re}} & -\frac{t}{\mathrm{Re}} & \frac{2\beta}{r^2\mathrm{Re}} \end{bmatrix} \tag{2-17}$$

$$[M_1] = \begin{bmatrix} -\frac{\beta}{r^2}T - \frac{3\beta^3}{r^4\mathrm{Re}} + \frac{\beta}{r^4\mathrm{Re}} + \frac{\alpha^2\beta}{r^2\mathrm{Re}} & \frac{4\alpha\beta}{r^3\mathrm{Re}} & \frac{\alpha}{r}T + \frac{3\alpha\beta^2}{r^3\mathrm{Re}} - \frac{3\alpha}{r^3\mathrm{Re}} - \frac{\alpha^3}{r\mathrm{Re}} \\ -\alpha T - \frac{\alpha}{r^2\mathrm{Re}} & \frac{\beta^2}{r^3\mathrm{Re}} - \frac{\alpha^2}{r\mathrm{Re}} & -\frac{\beta}{r}T - \frac{\beta}{r^3\mathrm{Re}} \end{bmatrix} \tag{2-18}$$

$$[M_0] = \begin{bmatrix} -\frac{\beta}{r}tT + \frac{4\beta^3}{r^5\mathrm{Re}} & 2\frac{\alpha\beta}{r^2}T - \frac{4\alpha\beta}{r^4\mathrm{Re}} - \frac{2\alpha\beta t}{r^2\mathrm{Re}} & \alpha\left(t - \frac{1}{r^2}\right)T + \frac{\alpha t}{r^2\mathrm{Re}} + \frac{3\alpha}{r^4\mathrm{Re}} \\ 4\frac{\beta^2}{r} - \frac{2\alpha\beta^2}{r^3\mathrm{Re}} & tT + \frac{\alpha^2}{r^2\mathrm{Re}} - \frac{\beta^2}{r^4\mathrm{Re}} & -\frac{\beta}{r^2}T - 4\alpha\beta + \frac{\beta}{r^4\mathrm{Re}} + \frac{2\alpha^2\beta}{r^2\mathrm{Re}} \end{bmatrix} \tag{2-19}$$

$$[N_2] = \begin{bmatrix} -\frac{\beta}{r} & 0 & \alpha \\ 0 & 0 & 0 \end{bmatrix} \tag{2-20}$$

$$[N_1] = \begin{bmatrix} -\frac{\beta}{r^2} & 0 & \frac{\alpha}{r} \\ -\alpha & 0 & -\frac{\beta}{r} \end{bmatrix} \tag{2-21}$$

$$[N_0] = \begin{bmatrix} -\frac{\beta t}{r} & \frac{2\alpha\beta}{r^2} & \alpha t - \frac{1}{r^2} \\ 0 & t & -\frac{\beta}{r^2} \end{bmatrix} \tag{2-22}$$

where

$$T \equiv \alpha U - \frac{t}{Re} = 2\alpha(1-r^2) - \frac{1}{Re}(\alpha^2 + \frac{\beta^2}{r^2}) \tag{E-48}$$

and

$$t \equiv \alpha^2 + \frac{\beta^2}{r^2} \tag{E-34}$$

Since the parameter β, which determines the angular periodicity, only has discrete values as determined by the boundary conditions in Appendix G,

$$\beta = ni, \quad n=0,1,2,... \tag{G-4}$$

each value of β is considered independently. For $\beta=0$, it is found from the coefficients (2-15) through (2-22) that the equations uncouple, even more completely than in the cartesian coordinate case. In the first equation represented by these coefficients (the top row of the matrices), both $g(r)$ and $f(r)$ drop out. The coefficients remaining represent a fourth-order ordinary homogeneous differential equation in $h(r)$. The boundary conditions for this problem are derived in Appendices F and G. Taking $f(r)$ to be identically zero, the boundary conditions at the wall are

$$h(1) = 0 \tag{F-15b}$$

$$Dh(1) = 0 \tag{F-15c}$$

The boundary condition at $r=0$ from (G-40) is that all even derivatives of h are zero. In the second equation $h(r)$ drops out because all coefficients contain β. With f set identically to zero, this equation becomes a homogeneous,

second order differential equation in $g(r)$. The boundary condition at $r=1$ is

$$g(1) = 0 \tag{F-15a}$$

The boundary condition at $r=0$ from (G-40) is that $g(0)$ and all even derivatives of g are zero at the origin.

The consequence of the complete separation of the components g and h into two separate homogeneous differential equations is that there are two sets of eigenvalues which are derived independently of one another. The velocity in terms of f, g, and h is derived in Appendix F. With $\beta=0$ and $f(r)=0$, equations (F-13) become

$$u(r) = \frac{1}{r} D(rh(r)) \tag{2-23a}$$

$$v(r) = \alpha h(r) \tag{2-23b}$$

$$w(r) = \alpha g(r) \tag{2-23c}$$

Thus the axial and radial components are related to the function $h(r)$, while the angular velocity is completely independent.

Both of these homogeneous differential equations can be solved as eigenvalue problems. This technique is discussed in the section on numerical methods. From the eigenvectors of f and g the velocity and vorticity perturbation forms may be found. This was not done in this paper.

If $n=1$, that is, if $\beta = \beta_R + i$, then the two equations represented by equations (2-15) to (2-22) do not uncouple and must be solved simultaneously, with two of the components of the velocity vector potential as the unknowns. From Appendix F it may be observed that the most complicated

boundary condition at the wall

$$\beta f(1) = \alpha h(1) \tag{F-17a}$$

arises from setting g(r) equal to zero. From Appendix G it is also found that one of the boundary conditions for n=1 at r=0 is

$$g(0) = ih(0) \tag{G-41}$$

These inhomogeneous boundary conditions may be avoided by the choice of setting h(r)=0.

The result is that the first and second columns of the matrices of equations (2-15) through (2-22) give the coefficients of f and g for each of the two equations (the top and bottom rows of the matrices). The boundary conditions at the walls are

$$g(1) = 0 \tag{F-19a}$$

$$f(1) = 0 \tag{F-19b}$$

$$Df(1) = 0 \tag{F-19c}$$

The boundary conditions at r=0 are that f(0) and all even derivatives of f are zero, and that g(0) and all odd derivatives of g are zero.

Thus, the problem for n=1 becomes two simultaneous homogeneous differential equations which must be solved together. This problem is discussed in the section on numerical methods.

D. STABILITY CRITERION

Previous investigations into the question of the stability of Poiseuille flows have used various criteria for determining the stability of the flow from the solutions

obtained. In particular, the real part of the eigenvalue γ_R is usually used when there is no real-exponential spatial variation [Salwen and Grosch 1972]. Where exponential growth in space has been a part of the problem [Garg and Rouleau 1971] the real part of the spatial wave number has been taken to give the instability. In other cases, the phase velocity has been used to determine the stability. Clearly, a measure of the stability of the flow must take into account both the exponential rate of growth with time and with space.

Since it is the stability of the perturbed fluid particles that is of concern, it is natural to consider how the rate of growth or decay of the perturbations is seen from a coordinate frame moving with a particular fluid particle. For cartesian coordinates, the fluid particles can have velocities ranging from 0 to 1.5 depending on their distance from the walls, that is, depending on the value of y. The distribution of fluid velocities is the Poiseuille distribution of equation (2-3).

$$U(y) = \frac{3}{2}(1-y^2) \tag{2-24}$$

These moving axes will have a velocity with the mean flow in the x direction. Let x', y, z, t be the coordinates and α, β, γ' the complex wave numbers with respect to the moving axes. The form of the perturbation vector potential for a given eigenvalue obtained as a solution is

$$\bar{W} = (\bar{j}g(y) + \bar{k}h(y))\exp(\alpha x + \beta z + \gamma t). \tag{2-25}$$

The complex frequency γ' seen from this moving reference frame will be different than γ from the fixed frame. To establish the relation between γ' and γ, the perturbation is written in the moving frame and then transformed into the form which corresponds to the fixed frame.

$$\bar{W} = (\bar{j}g + \bar{k}h)\exp(\alpha x' + \beta z + \gamma' t)$$

$$= (\bar{j}g + \bar{k}h)\exp[\alpha(x-Ut) + \beta z + \gamma' t]$$

$$= (\bar{j}g + \bar{k}h)\exp[\alpha x + \beta z + (\gamma' - \alpha U)t]$$

$$= (\bar{j}g + \bar{k}h)\exp[\alpha x + \beta z + \gamma t] \tag{2-26}$$

Thus

$$\gamma' - \alpha U = \gamma \tag{2-27}$$

Solving for γ' and splitting into real and imaginary parts gives

$$\gamma'_R = \gamma_R + \alpha_R U \tag{2-28}$$

$$\gamma'_I = \gamma_I + \alpha_I U \tag{2-29}$$

If γ'_R is positive, zero, or negative, the perturbation is said to be unstable, neutral, or stable, respectively. Hence, the value of γ'_R is taken to be a measure of the stability of each eigenvalue obtained.

III. NUMERICAL METHODS

A. CARTESIAN COORDINATES

The equation to be solved is the linear combination of the first and third components of the vorticity transport equation which uncouples the dependence of $h(y)$ and $g(y)$. This is developed in Section II and results in equation (2-13).

$$-\frac{\alpha}{Re}D^4h + \left(\alpha T - \frac{\alpha}{Re}(\alpha^2+\beta^2)\right)D^2h + \left(3\alpha^2 + \alpha(\alpha^2+\beta^2)T\right)h$$
$$+ \gamma\left(\alpha D^2h + \alpha(\alpha^2+\beta^2)h\right) = 0 \qquad (2\text{-}13)$$

where T is defined by equation (D-54).

$$T \equiv \alpha U - \frac{1}{Re}(\alpha^2+\beta^2) = \frac{3}{2}\alpha(1-y^2) - \frac{1}{Re}(\alpha^2+\beta^2) \qquad (D\text{-}54)$$

The boundary conditions are derived in Appendix F.

$$h(\pm 1) = 0 \qquad (F\text{-}4b)$$

$$Dh(\pm 1) = 0\ . \qquad (F\text{-}4c)$$

Various methods are applicable to the solution of this homogeneous fourth-order equation. The method of solution applied is the method of finite differences, approximating the function $h(y)$ by n discrete evenly spaced unknowns, and solving for the eigenvalues and eigenvectors of the system of equations thus generated.

One of the variables, α, β or γ must be chosen to be the unknown eigenvalue. Since γ is linear while α and β occur to higher powers, the choice of γ produces a more standard problem. Thus, γ is the unknown and values must be assigned to α and β. Equation (2-13) has already been written in the appropriate form for this method of solution, with the factors of γ separated from the rest of the equation. In

an abbreviated, but general, form, equation (2-13) can be written

$$C_4 D^4 h + C_2 D^2 h + C_0 h + \gamma(C_2' D^2 h + C_0' h) = 0 \qquad (3\text{-}1)$$

where the coefficients (C's) are, in general, functions of α, β, and y.

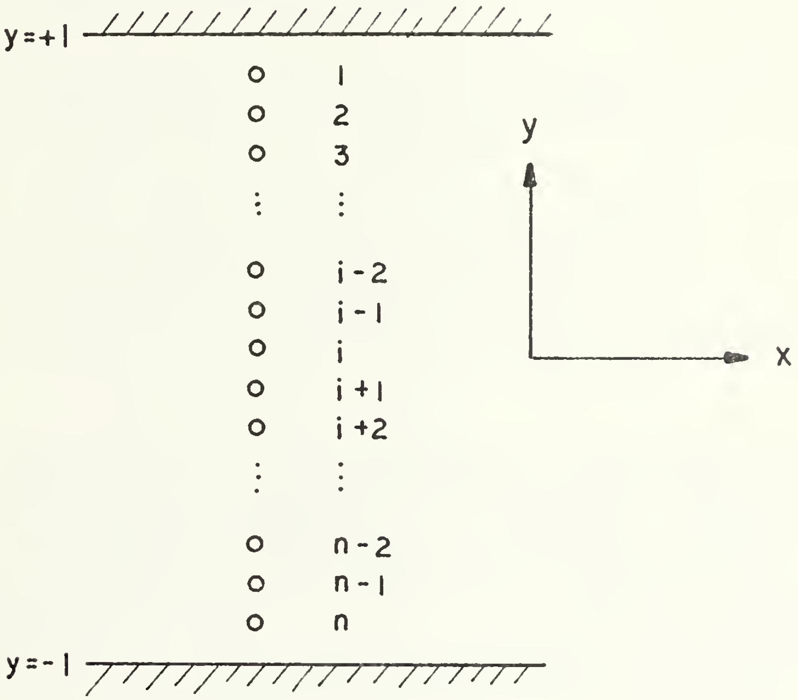

Figure 3-1 Finite Difference Mesh

The range of y, from +1 to -1 across the channel height, is divided into a one-dimensional computational mesh of even spacings. As shown in Figure 3-1, there are n interior points, n+1 divisions and n+2 total mesh points, including the boundaries. The grid spacing is δy.

If the values for the function h(y) at the grid points i-2, i-1, i+1 and i+2 are expanded as Taylor series about the i^{th} grid point, the resulting set of simultaneous

equations may be solved for the second-order central difference approximations of the derivatives. These are as follows.

$$Dh_i = (h_{i+1} - h_{i-1})/2(\delta y) \qquad (3\text{-}2a)$$

$$D^2h_i = (h_{i+1} - 2h_i + h_{i-1})/(\delta y)^2 \qquad (3\text{-}2b)$$

$$D^3h_i = (h_{i+2} - 2(h_{i+1} - h_{i-1}) - h_{i-2})/2(\delta y)^3 \qquad (3\text{-}2c)$$

$$D^4h_i = (h_{i+2} - 4h_{i+1} + 6h_i - 4h_{i-1} + h_{i-2})/(\delta y)^4 \qquad (3\text{-}2d)$$

The error of equations (3-2) is of the order of magnitude $(\delta y)^2$.

The values of h at the boundaries $h(\pm 1)$ are known to be zero by the boundary condition of equation (F-4b). Thus, there are n unknowns--the values of h(y) at the n interior points--denoted h_1, h_2, ..., h_i, ..., h_n. By substituting equations (3-2) for the derivatives of h in equation (2-13), the fourth order differential equation becomes a set of n linear, algebraic, difference equations which form a banded matrix with a band width of five. Each equation, written for the i^{th} mesh point, includes the values of h at the mesh points i-2, i-1, i, i+1, and i+2. There are four special cases which must be handled using the boundary conditions. The equations written for the points 2 and n-1 include the boundaries. Since the values for h at the boundaries is known to be zero, those terms drop out of those equations. The equations written for points 1 and n contain terms for the boundaries and terms for virtual points outside the boundaries, which, in accordance with the labeling scheme,

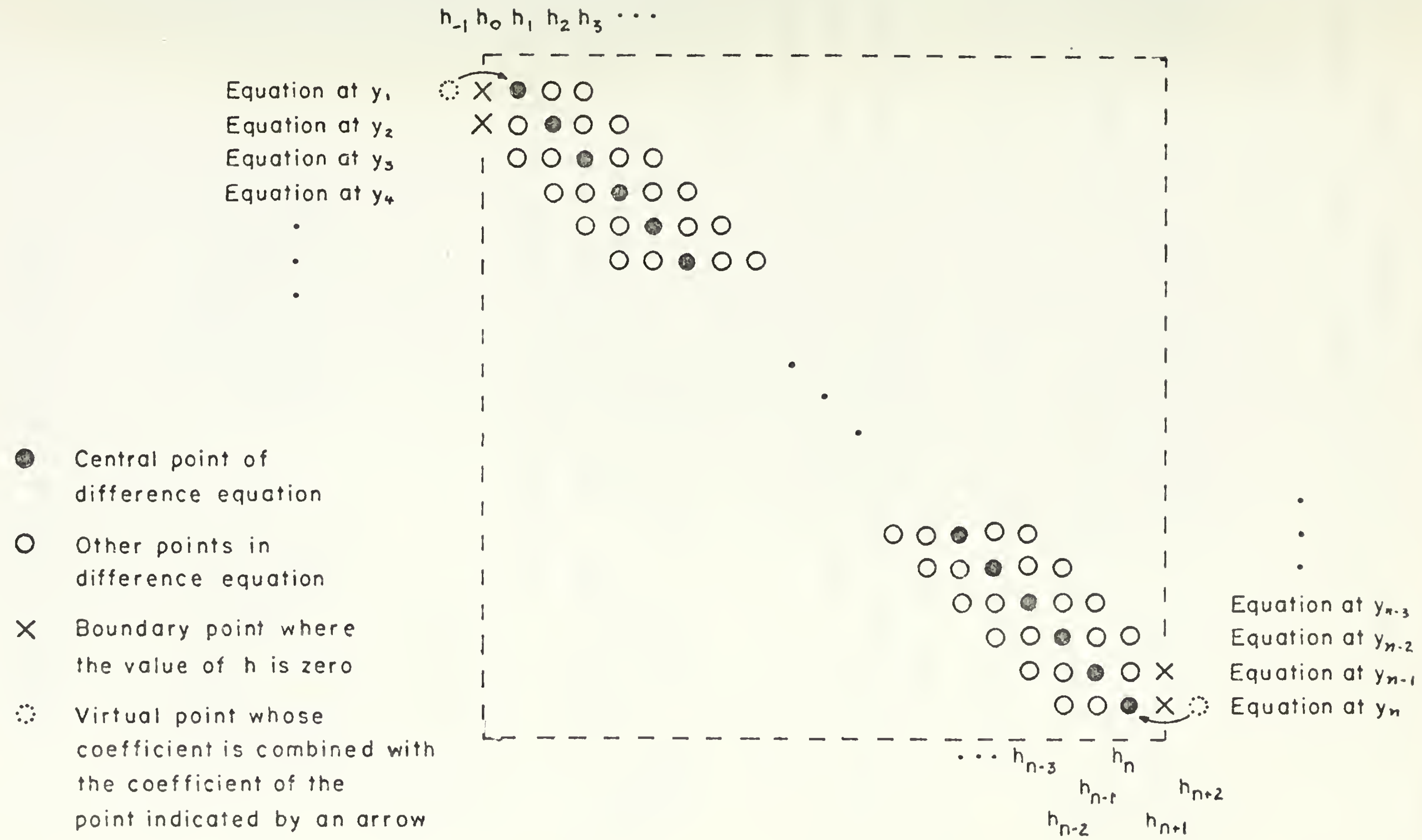

Figure 3-2 Finite Difference Matrix

may be called the -1 and $n+2$ values of h (see Figure 3-2). The second boundary condition, applied at the zeroth (boundary) point, yields the condition

$$h_{-1} = h_{1} \; . \tag{3-3}$$

Thus, in the difference equation written at the first interior point, the term for the boundary is dropped and the coefficient of the virtual point is combined with the coefficient at the first interior point. The equation written for the n^{th} point is treated similarly.

The matrix thus formed is broken into two matrices, one containing the coefficients of the eigenvalue γ, and the other containing the remaining terms. The resulting matrix equation may be written

$$[X]\{V\} - \gamma[Y]\{V\} = 0 \tag{3-4}$$

Matrix [Y] will be a three-banded matrix because γ only occurs as a factor of D^2h and h. $\{V\}$ is the eigenvector of values of the function h at the mesh points. Figure 3-3 shows the structure of the matrices.

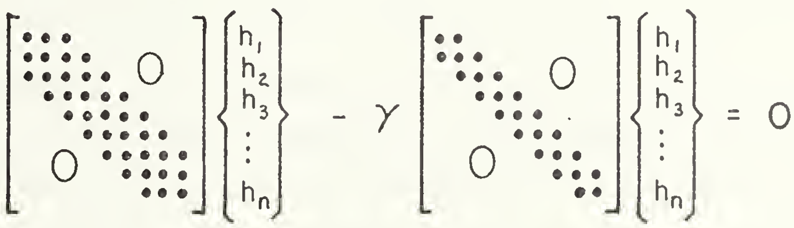

Figure 3-3 Matrix Form of Equation

The two subprograms, FUNCTION CHM1E1 and FUNCTION CHM2E1 supply the coefficients of these matrices. The input parameters, K and Y are, respectively, the relative position in the band for each row, and the value of y for the difference equation. The subprograms contain function statements which produce the coefficients obtained from equation (2-13). The parameter K determines how those coefficients are combined to yield the coefficients of the points i-2, i-1, i, i+1, and i+2, which are determined using the central difference equations, (3-2). Because the standard eigenvalue equation has a minus sign, as in equation (3-4), the signs of the coefficients of CHM1E1 are reversed from the sign of the terms in equation (2-13).

The eigenvectors obtained from this problem have either even or odd symmetry, that is, they have symmetry about $y=0$ such that either

$$h(y) = h(-y) \tag{3-7a}$$

or

$$h(y) = -h(-y) \ . \tag{3-7b}$$

Using this fact, the problem size can be cut in half by solving for the values of h_i across only half of the channel, once with boundary conditions at $y=0$ corresponding to even symmetry and a second time with boundary conditions corresponding to odd symmetry. This gives the full set of eigenvalues and vectors. The boundary conditions for the mesh points near the center of the channel are easily derived from the conditions in equation (3-7).

Subroutine MSET sets up the coefficients of either matrix X or matrix Y. The variable MODE is used to control how the matrices are set up. If MODE=0, the matrices for

the full channel result. MODE=1 or MODE=2 will result in matrices set up for the half channel with boundary conditions for either odd or even symmetry. For the half-channel solution, the total number of divisions in the channel is always even so that the center of the channel, y=0, falls between two mesh points. The center of the channel could fall directly on a mesh point but this would result in the inconvenience of having one more eigenvalue in the solution for even vectors than in the solution for odd vectors.

At the time of writing there were no computer routines available to solve an eigenvalue problem with complex matrices in the form of equation (3-4). Therefore, the problem is converted to the more conventional form

$$([Z] - \gamma[I])\, V = 0 \tag{3-8}$$

by inverting matrix Y and forming the product

$$[Z] = [Y][X]^{-1} \tag{3-9}$$

The subroutine DEIGEO solves the matrix problem of equation (3-4). Subroutine MSET is called to set up matrix X. CDMTIN is then called to invert [X]. CDMTIN was obtained by modifying the IBM library routine CMTRIN to make it applicable to double precision matrices. Next, BMSET is called to set up matrix Y using space-saving band storage. The two matrices are multiplied using subroutine MULDBM. Since all the programs available for solving the resulting equation require that the real and imaginary part of the matrix be presented in separate matrices, subroutine DSPLIT is called to accomplish this. The programs used to solve for the eigenvalues are EHESSC and ELRH1C which are available through the International Mathematical and Statistical Library. These subroutines reduce the matrix

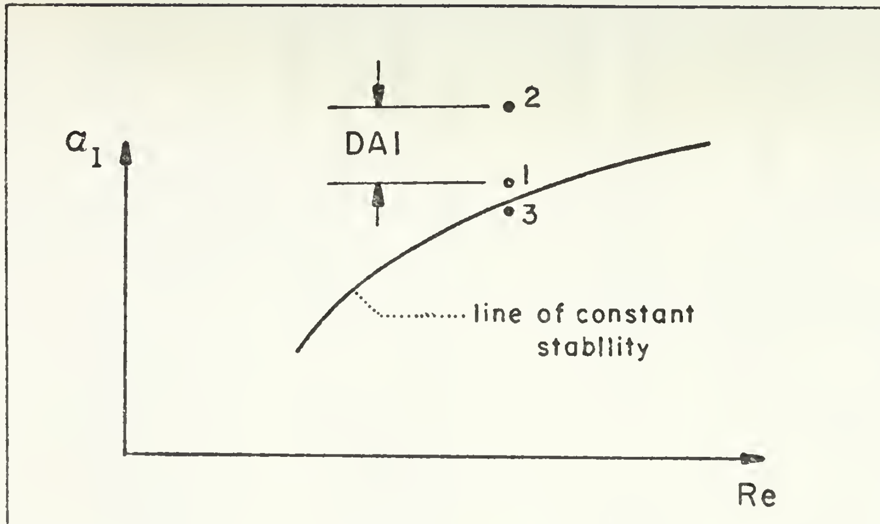

Figure 3-4a, Program 2, Mode = 2

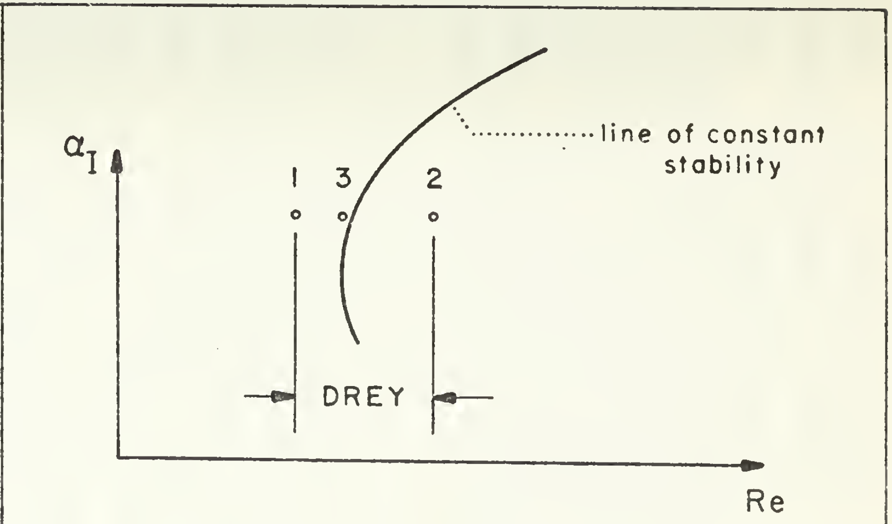

Figure 3-4b, Mode = 3

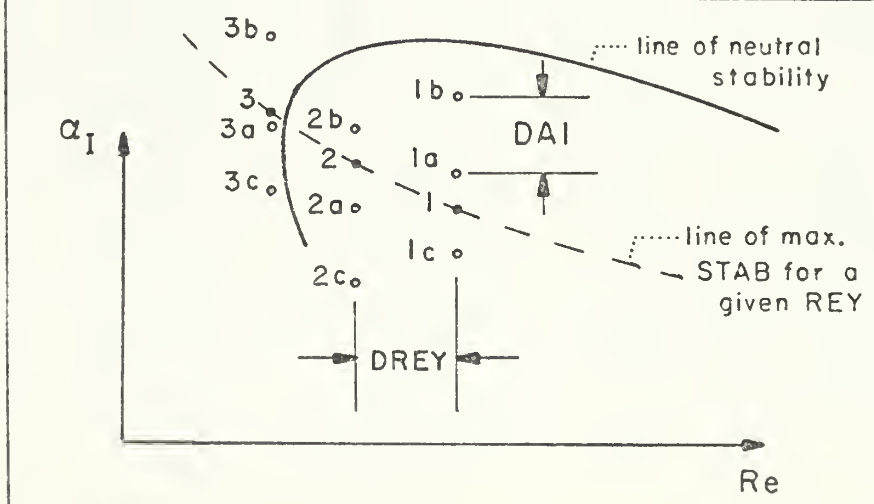

Figure 3-4c, Mode = 4

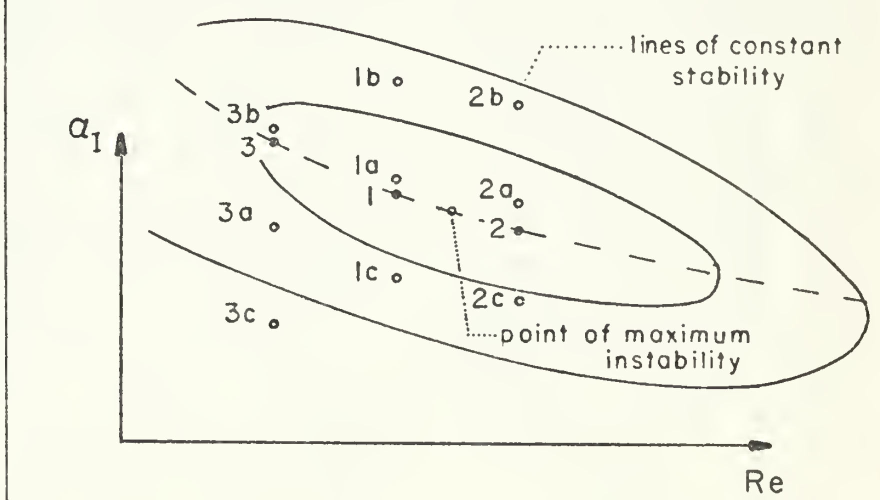

Figure 3-4d, Mode = 5

into the complex upper Hessenberg form and then solve for the eigenvalues. Another program which can be used to solve for the eigenvalues is EISPAC (Eigensystem Subroutine Package) which is available from the IBM library. This program was also used to solve this problem and gave results identical to the results obtained using the I.M.S.L. routines. DEIGEO solves the problem twice, once for the even eigenvectors and once for the odd eigenvectors. The results are then passed back to the calling program.

Three main programs have been written. Program #1 takes values of α and β as input, calls DEIGEO, and outputs a list of the eigenvalues, along with the stability for each eigenvalue, determined using equation (2-35). A printer plot is also output. Program #1A is the same as program #1 except that a plot on the Calcomp plotter is output using subroutine DRAW. This output is used in this paper to present the graphs of the eigenvalues. Program #2 was used to determine the shape of the stability curves. This program performs five functions depending on the value of the control parameter MODE. The stability curves determined are all two-dimensional plots of α_I (the imaginary part of α) versus the Reynolds number, holding β_I, β_R, α_R, the velocity, and the stability constant.

For MODE=1, the stability for a given set of input parameters, α, β, Reynolds number (REY) and velocity (VEL) is determined and printed. Equation (2-35) is used to determine the stability.

For MODE=2 a point on the line of constant stability with an input value of STAB is determined for a given input value of REY. Referring to Figure 3-4a, the initial guess

has a value of α_I and REY, and corresponds to point 1. A second point (2) is determined by adding DAI to the initial value of α_I. The stability at both of these points is determined by calling the subroutine EIGLS which determines the value of the stability of the least stable eigenvalue for a given set of input parameters, α, β, REY, and VEL. A linear interpolation on α_I is made to determine a third guess (3) for which the stability is also calculated. Subroutine DFIT2 is then called to make a second-order approximation of the value of α_I which corresponds to the desired value of the stability (STAB). This final value is printed.

For MODE=3, the same operations are performed as for MODE=2, except that REY is varied instead of α_I. Figure 3-4b shows this. The initial guess is point 1, DREY is used to find point 2, a linear interpolation to find 3 and a second-order fit to determine a final estimate, which is printed.

For MODE=4, the point of minimum REY on the neutral stability curve is determined. Two points (1b and 1c) are determined from the initial guess (1a), by adding and subracting DAI from α_I. Their stabilities are determined by calling EIGLS and a second order fit determines the least stable point (1) for the initial value of REY. A new value of REY is picked by adding or subracting DREY as appropriate. This results in point 2a. The points 2b and 2c are found by adding ±DAI. The stability of these three points is determined, resulting in the least stable point

(2) for this value of REY. A third point (3a) is determined using a linear interpolation from points 1 and 2, two points (3b and 3c) are chosen by adding ±DAI and the stabilities of these points results in the least stable point (3). Finally, a second-order fit is made using points 1, 2, and, 3 to find the point corresponding to zero stability. This point will be the point of minimum Reynolds number on the neutral stability curve.

For MODE=5, the point of minimum stability is determined for a set of input values α_R, β_I, β_R, and VEL. The procedure is similar to that for MODE=4. An initial guess (1a) and two points (1b and 1c) are used to find point 1. This procedure is repeated twice at values of REY determined by adding or subtracting DREY from the initial guess. Points 1, 2, and 3 are then used to approximate the point of minimum stability using subroutine DFIT2.

B. CYLINDRICAL COORDINATES

Programs to solve the two cases, n=0 and n=1, were written separately. The method of finite differences is used in each case. All subroutines used in the cartesian case are used in the cylindrical case with the exceptions of the functions returning the coefficients and the subroutine setting up the matrices.

Functions CHM1E1, CHM2E1, CHM1E2, and CHM2E2 return the values of the coefficients of the velocity vector potential component h for the five adjacent mesh points in the central difference representation of the equations. This is done in a manner exactly the same as the functions CHM1E1 and CHM2E1 in the cartesian case. The "M2" means those terms which also contain the eigenvalue γ, and are used to set up the matrix Y. The "M1" means those remaining terms which do not

contain γ, and are used to set up the matrix X. "E1" means that the terms come from the linear combination of Γ_x and Γ_θ given in equation (2-14) and which are written out as the top row of the matrices of (2-15) through (2-22). "E2" means that the terms come from the equation Γ_r, and which are the bottom row of the matrices of (2-15) through (2-22). Similarly, functions CGM1E1, CGM2E1, CGM1E2, and CGM2E2 return all the values of the finite difference coefficients of the component g(r), and CFM1E1, CFM2E1, CFM1E2, and CFM2E2 return the values of the coefficients of f(r).

Subroutine MSET2 performs exactly the same function as MSET, with the exception that MSET2 allows the matrix to be positioned with a starting value other than (1,1). This is necessary because for cases other than n=0, two equations must be solved simultaneously and matrices must be placed adjacent to each other to accomplish this. Also, MSET2 forms a mesh which includes the point r=0, in contrast to MSET which had the center of the channel fall between mesh points. The boundary conditions must be written appropriately. MSET2 sets the correct boundary conditions at the wall r=1, but does not set any boundary conditions at r=0 since they are different for each case. Thus, the boundary conditions at r=0 must be set by the calling program.

1. Axisymmetric Flow (n=0)

Program #R1A solves the first equation for the function h(r). As shown in part II, section C, the boundary conditions can be written

$$h(1) = 0 \tag{3-10}$$

$$Dh(1) = 0 \tag{3-11}$$

$$h(0) = 0 \tag{3-12}$$

$$D^2h(0) = 0 \tag{3-13}$$

All even derivatives at the origin are zero, but only (3-12) and (3-13) are needed to satisfy the difference equations at r=0. The above equations imply that the boundary mesh points at r=0 and r=1 are left out of the difference equations. When the virtual point outside the boundary r=1 appears, its coefficient is added to the first mesh point inside the boundary. When the virtual point outside the boundary r=0 appears, the negative of its coefficient is added to the first mesh point inside the boundary. The boundary conditions at r=1 are set by subroutine MSET2 while the boundary conditions at r=0 are set by the main program #R1A. The rest of the solution proceeds exactly the same as the method used for the cartesian case discussed in section A of part III.

Program #R1B solves the second equation for the function g(r). The boundary conditions, from part II section C, are

$$g(1) = 0 \tag{3-14}$$

$$g(0) = 0 \tag{3-15}$$

In addition to (3-15), all odd derivatives of g are zero at r=0, but only (3-15) is required to satisfy the difference equations at r=0 since the highest derivative of g is the second derivative and the result is a tri-diagonal matrix. Again, the solution procedure is identical to the cartesian case.

Both programs #R1A and #R1B output a list of the eigenvalues and the stability for each eigenvalue. Printer plots of the eigenvalues are also output.

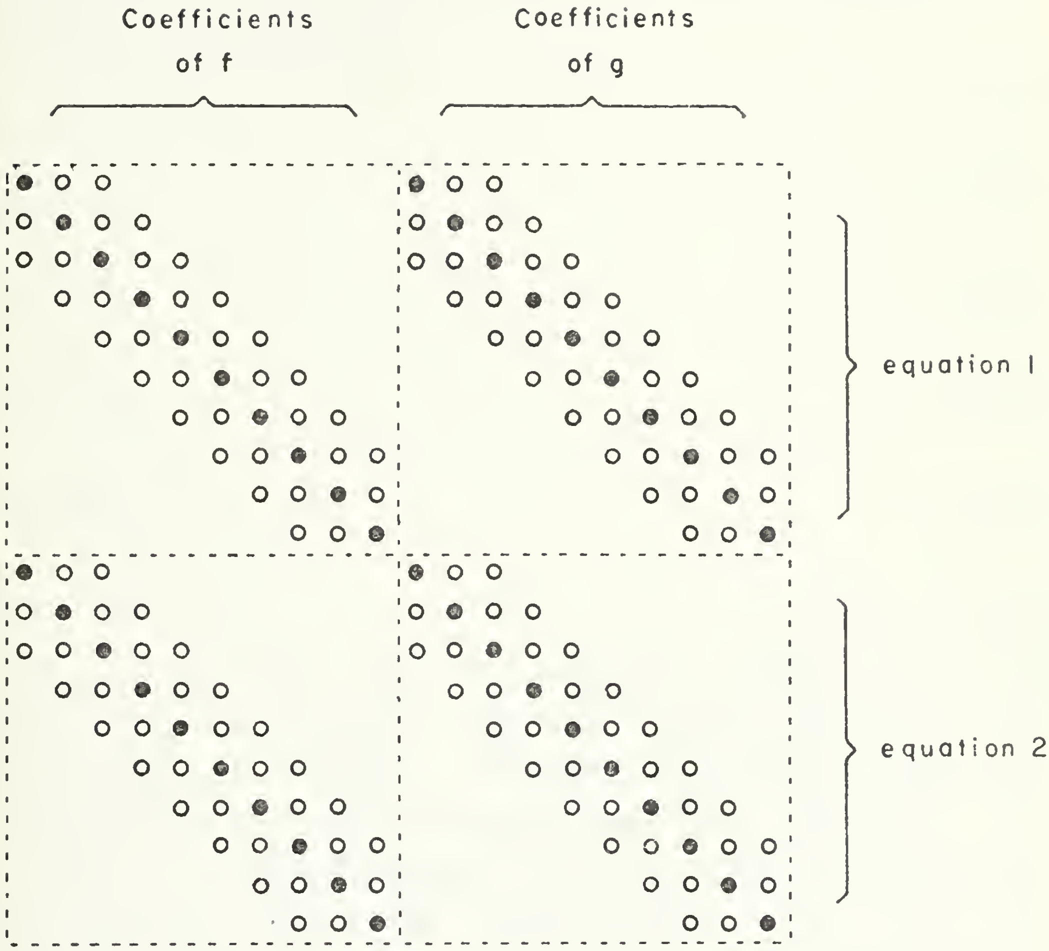

Figure 3-5 Finite Difference Matrix for Pipe Flow

2. n=1

For this case the two equations must be solved simultaneously. This is done by writing two sets of n equations for the set of 2n variables including n unknown values of f and n unknown values of g. The function h is set identically to zero as described in section II, part C. The boundary conditions are

$$f(1) = 0 \tag{3-16}$$

$$Df(1) = 0 \tag{3-17}$$

$$g(1) = 0 \tag{3-18}$$

$$f(0) = 0 \tag{3-19}$$

$$D^2f(0) = 0 \tag{3-20}$$

$$g(0) = 0 \tag{3-21}$$

The matrices are set up as shown in figure 3-5 using subroutine MSET2 four times. The upper left section is set up using function CFM1E1. CGM1E1 is used for the upper right section, CFM1E2 for the lower left and CGM1E2 for the lower right. A second matrix is set up using an identical procedure, but for the "M2" coefficients, that is, the coefficients which contain γ. The procedure is then to invert this second matrix, multiply it with the first and solve for the eigenvalues. Program #R2 carries out the solution and then lists the eigenvalues found, with their stability. A plot is also output.

IV. RESULTS

A. COMPARISON WITH PREVIOUS RESULTS

1. Neutral Stability Curve

The neutral stability curve was calulated for the two-dimensional, sinusoidal case, that is, for $\alpha_R = \beta_I = \beta_R = 0$. Figure 4-1 compares the results with those of Grosch and Salwen (1968). Using n=60 (60 mesh points across the half channel) gives almost the same accuracy as the published results. The computation time for finding the eigenvalues using n=60 was over one minute (on an I.B.M. 360), compared to a computation time of approximately 10 seconds for n=30. For this reason a mesh size of 30 was used to obtain all results presented in this paper except where noted.

2. Squire's Theorem

Squire's theorem can be verified numerically. As expected, the rotation of three-dimensional sinusoidal disturbances onto the two-dimensional plane always has a stabilizing effect. From Squire's theorem, it follows that the addition of a transverse sinusoidal component to a two-dimensional sinusoidal wave is absolutely stabilizing in that the minimum Reynolds number on the neutral stability curve must increase for increasing β_I. Figure 4-12, which plots the minimum Reynolds number for various values of α_R and β_I, shows this to be so. For $\alpha_R=0$, the neutral stability curve does not seem to have another lobe (labelled e-f-g in figure 4-10a), as the curves do for $\alpha_R \neq 0$. For the cases for which Squire's theorem applies, $\alpha_R=0$, figure 4-12

shows that increasing β_I is stabilizing. Squire's theorem is not applicable to the other cases, where $\alpha_R \neq 0$.

B. STABILITY CURVES

The stability criterion used is discussed in section II-D. It is based on how an observer in a moving coordinate system would experience the combination of exponential growth in space and in time of the perturbations. The stability is given by γ'_R in equation (2-28).

$$\gamma'_R = \gamma_R + \alpha_R U \qquad (2\text{-}28)$$

The stability observed from a coordinate system moving with velocity U is said to be stable, neutral or unstable according to whether γ'_R is less than, equal to, or greater than zero. The velocities of interest are those of the fluid particles, which vary from 0 to 1.5.

The neutral stability curves for different values of α_R, with $\beta_I = 0$, are shown in figure 4-2. The stability criterion used is for a fixed coordinate system where U=0 in equation (2-28). This is the same as the instability observed from a nonmoving frame of reference. The structure of the curves for $\alpha_R \neq 0$ appears to be qualitatively different from the curve for $\alpha_R = 0$ in that (referring to figure 4-10a) the lobe e-f-g is not present on the neutral curves for $\alpha_R = 0$.

Figure 4-3 shows the effect on the neutral curves of the

addition of a β_I component. It appears that only the upper lobe (labelled b-c-d in figure 4-10a) of the curves is affected. The stability is γ'_R from equation (2-28), for a fixed coordinate system (U=0).

Cases involving nonzero values of β_R were not computed. This was primarily because of a lack of time, but also because a hypothetical case of this kind might well be difficult if not impossible to realize experimentally.

Curves of constant stability for $\alpha_R=0$ and $\alpha_R=-.04$ are shown in figures 4-4 and 4-5. It is important to note that equation (2-28) can be used (for nonzero values of α_R) to transform values of calculated stability as seen by one reference frame, into different values of stability as seen by another reference frame moving with a different U. Conversely, if a curve of constant stability is given for one reference frame, it is possible to find another reference frame which sees that curve as having any desired stability, such as neutral stability, for example. The velocity of this frame cf reference is meaningful only if it falls within the limits of the fluid particle velocities, 0 to 1.5. Thus, for $\alpha_R=-.04$, the neutral curve for a coordinate frame moving with the mean velocity, U=1.0, corresponds to the curve of constant stability for $\gamma_R=.04$ as seen from a fixed coordinate frame. Correspondingly, the neutral curve for the maximum velocity, U=1.5, corresponds to the curve of constant stability for $\gamma_R=.06$ for U=0.

C. EIGENVALUES

The method of finite differences was used to solve the differential equations (described in section III). A mesh of n finite difference grid points was placed across the half channel.

The effect of mesh size on the eigenvalues is demonstrated in figures 4-6a,b,c, and d. From these graphs it can be seen that the effect of increasing the accuracy of the solution by increasing the number of mesh points is to stretch out the eigenvalues. For high enough n, the eigenvalues form a "Y" lying on its side with the branches near the γ_I axis. The other two branches, which move away from the γ_I axis with increasing n, seem to be "false" eigenvalues which can be made ever increasingly stable by improving the accuracy of the solution. A true, qualitative picture of the eigenvalues is as shown in figure 4-10b, with the line of eigenvalues extending indefinitely to the left. Note also that in all the eigenvalue plots shown, the value of γ_I corresponding to the base of the "Y" formed by the eigenvalues is equal to the value of α_I.

The effect of Reynolds number is shown by figures 4-7a,b,c,d. It is apparent that changing Reynolds number has an effect similar to that of changing n. This seems to indicate that, for a given mesh size, lower Reynolds numbers yield more accurate eigenvalues than higher ones. This is shown to be true later in this section.

The effect of α_R on the eigenvalues is shown by the

series of plots in figures 4-7b, 4-8a, and 4-8b. The effect of decreasing α_R is to make the flow less stable. Note that both branches of the eigenvalue plots become unstable. No plots of positive α_R are shown, but the results are to shift the eigenvalues to decreasing γ_R, that is, stabilizing.

The effect of α_I is shown in figures 4-9a, 4-7b, and 4-9b. Increasing α_I tends to increase the frequency(γ_I), that is, shorter wavelengths are associated with higher frequencies.

The significance of the different parts of the stability curves is shown in the series of figures, 4-11. The neutral stability curve for α_R=-.04 is shown in figure 4-10a with the parts of the curve labelled. Figure 4-10b shows a typical eigenvalue plot for large n with the branches labelled. Eigenvalues for the points labelled A, B, C, D and E, are plotted in figures 4-11a,b,c,d,e. From these plots it is observed that the traditional neutral curve, represented by the instability of point B, is associated with branch LF of the eigenvalues corresponding to lower frequencies. The curve a-b d-e that cuts vertically through the upper lobe is associated with the instabilities of points A and D. These points have instabilities on the other branch (HF) of the eigenvalues, the one associated with higher frequencies. Figure 4-11c illustrates that at point C, both branches of the eigenvalues are unstable. At point E, figure 4-11e shows that the two branches, HF and LF, of the eigenvalues have almost disappeared, which seems to relate to the sudden decrease in stability of the lower lobe e-f.

D. STABILITY

The instability of the top lobes (labelled b-c-d in figure 4-10a) of the neutral curves of figures 4-2 and 4-3 is qualitatively different from instabilities elsewhere. For this reason, and because these lobes are directly analogous to the neutral curve for $\alpha_R=0$, this region of the curves is considered separately from the rest of the instabilities. Each of these lobes has a point of minimum Reynolds number. These are plotted in figure 4-12 for different values of α_R and β_I. Once again, the stability criterion is based on equation (2-28) for a fixed coordinate system.

Each of the upper lobes (b-c-d) also has a maximum instability associated with it similar to the maximum instability shown in figures 4-4 and 4-5. These values are shown in figure 4-13 which plots the coordinates (α_I vs. Re) of the maximum instabilities. Only the case of $\beta_I=0$ is considered. Note that the values of α_R plotted only go to -.04. This is because the instabilities associated with the part of the neutral curve labelled a-b d-e in figure 4-10a overwhelm the instabilities of the lobe b-c-d as α_R gets increasingly negative.

If the values of the maximum instabilities are plotted vs. α_R, as in figure 4-14, they form the curve labelled

$U=0$. The values of stability of figure 4-13 correspond to a nonmoving coordinate system. Using equation (2-28), the maximum instabilities for nonzero velocities are plotted. The lines appear to be almost straight. Note that for a velocity of approximately .41, the line of maximum instability is approximately horizontal. For velocities greater than .41, the line intersects the α_R axis at some negative value of α_R. For α_R below this value, the flow is stable. Assuming that these lines are straight, values of velocity less than .41 will make the flow unstable at Reynolds numbers as low as desired by choice of a suitably negative value of α_R. Although not yet verified numerically, it seems possible that for velocities greater than .41 there may be a critical value of the Reynolds number below which no instabilities are found. Curves of constant stability for positive α_R must be plotted to see if this is so. Cases of $\beta_I \neq 0$ were not considered due to lack of time.

The most critical portion of the neutral curve occurs at values of α_I approaching zero, that is, for long wavelengths. The stability of points along a line close to the Re axis (at a value of α_I=.01) was determined for values of α_R ranging from -.02 to -10.0. Equation (2-28) was used to translate these values of the stability found for a nonmoving coordinate frame of reference to the velocity of a coordinate axis for which the stability was exactly neutral. The resulting curves are plotted in figure 4-15. The region of instability lies below and to the right of the curves. It can be seen from the graph that there is no minimum

Reynolds number associated with these instabilities. In particular, a value of $\alpha_R=-10.0$ seems to be completely unstable for all velocities and Reynolds numbers.

E. PIPE FLOW

The results for the numerical solution of the pipe flow equations seem to indicate that there is an error somewhere in either the analysis or the numerical solution technique. All three programs, R1A, R1B and R2, produce eigenvalues which become increasingly unstable for decreasing Reynolds number. Further study is necessary to resolve this.

F. NUMERICAL ACCURACY

Figure 4-16 shows the neutral stability curves for two values of α_R for a mesh with 30 points. Portions of the curve with n=60 are also plotted. These graphs support the observation made earlier that decreasing Reynolds number seems to be associated with increasingly accurate solutions.

Calculations were done on an I.B.M. 360 computer in both single and double precision, that is, with computer word lengths of 32 and 64 bits respectively. Some calculations were also done on an X.D.S. 9300 with a 48 bit word length. With n=60, the eigenvalues obtained on the 9300 and in double precision on the 360 agreed to four to six places. Those obtained in single precision on the 360 sometimes agreed to only the first place with the double precision values. Thus it appears that the 64 or 48 bit word length is sufficient for matrices of size n=60 while in single precision on the I.B.M. 360, some eigenvalues have significant errors.

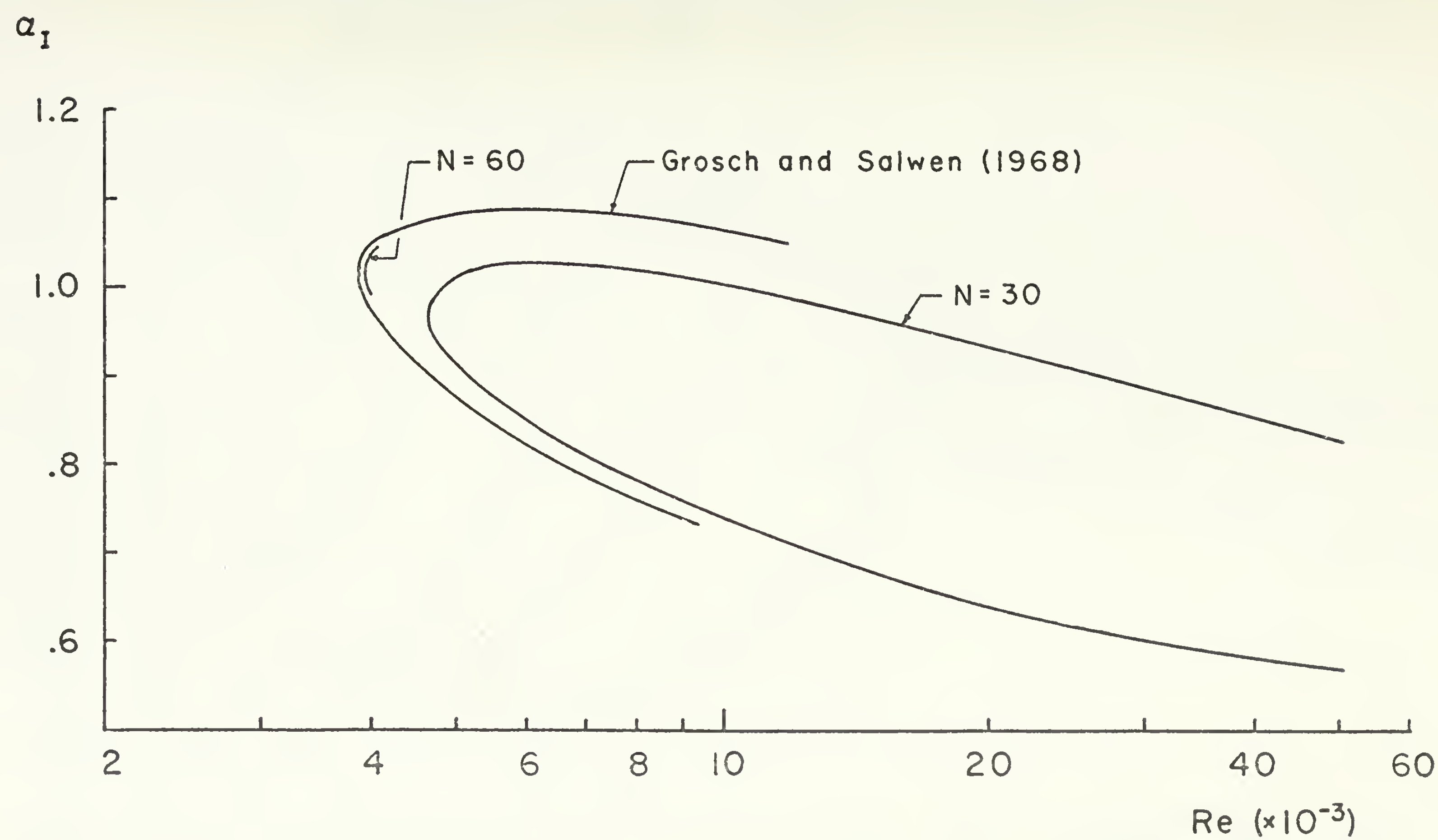

Figure 4-1 Comparison with previously published results.

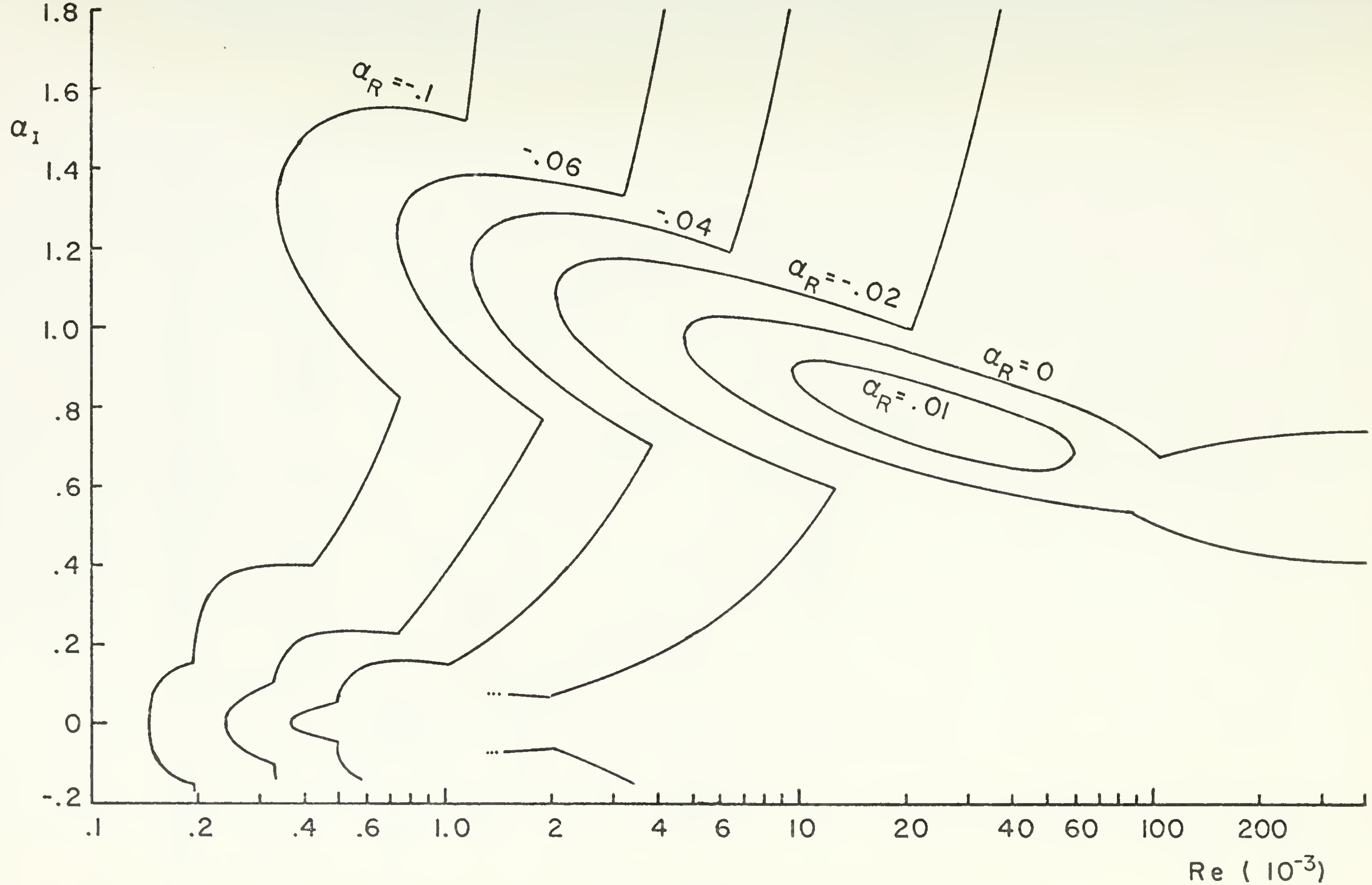

Figure 4-2 Neutral stability curves for various α_R

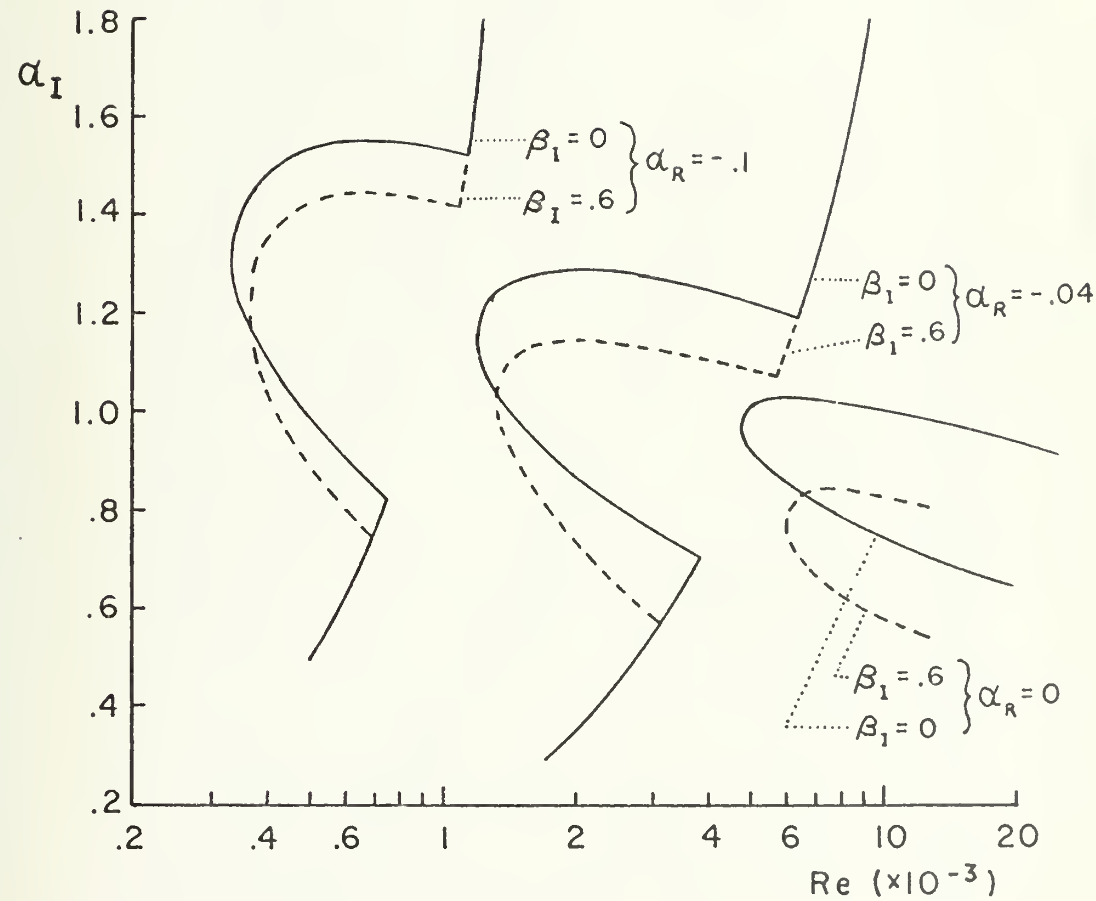

Figure 4-3 Neutral stability curves for various β_I

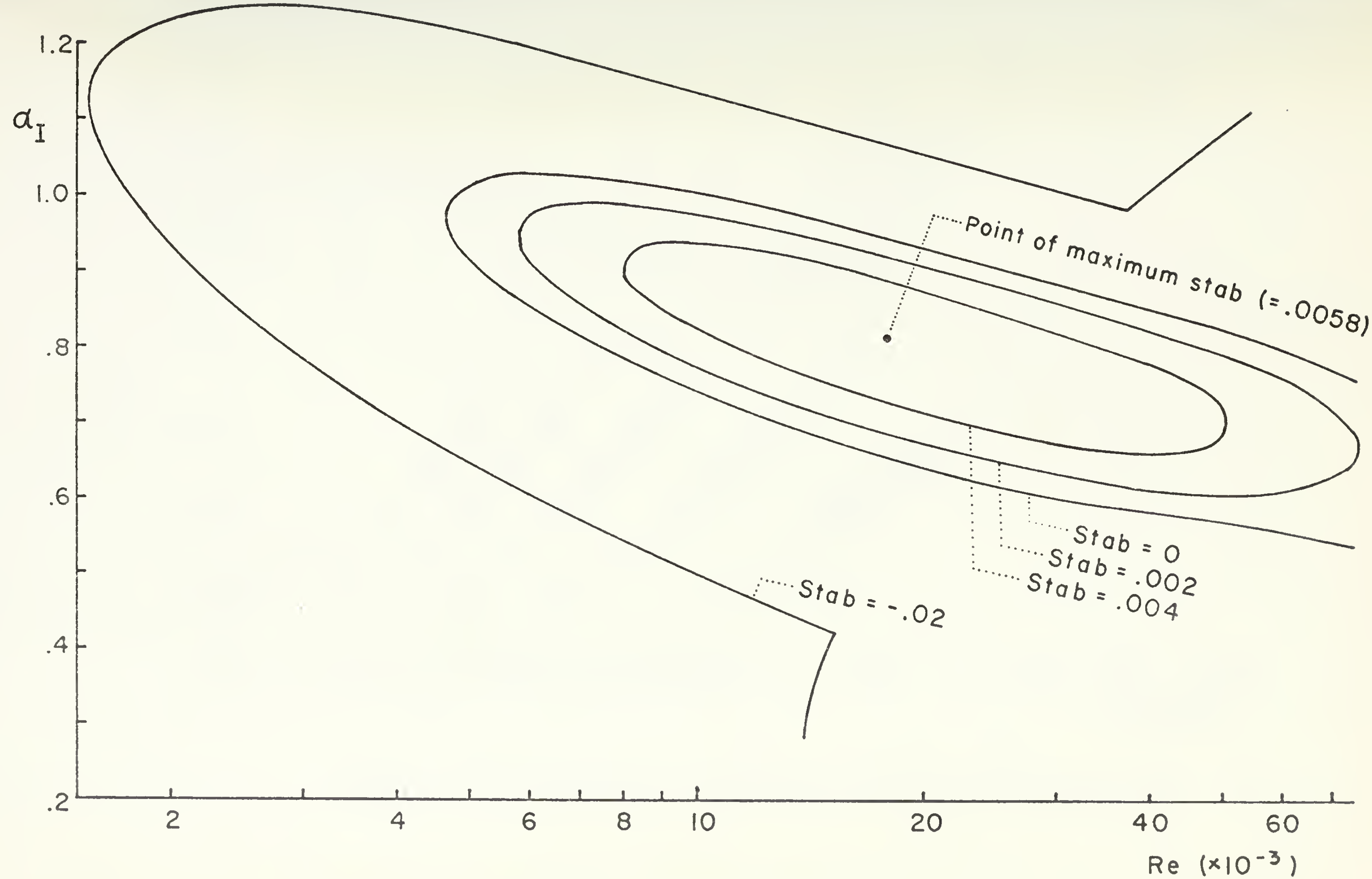

Figure 4-4 Curves of constant stability for $\alpha_R = 0$

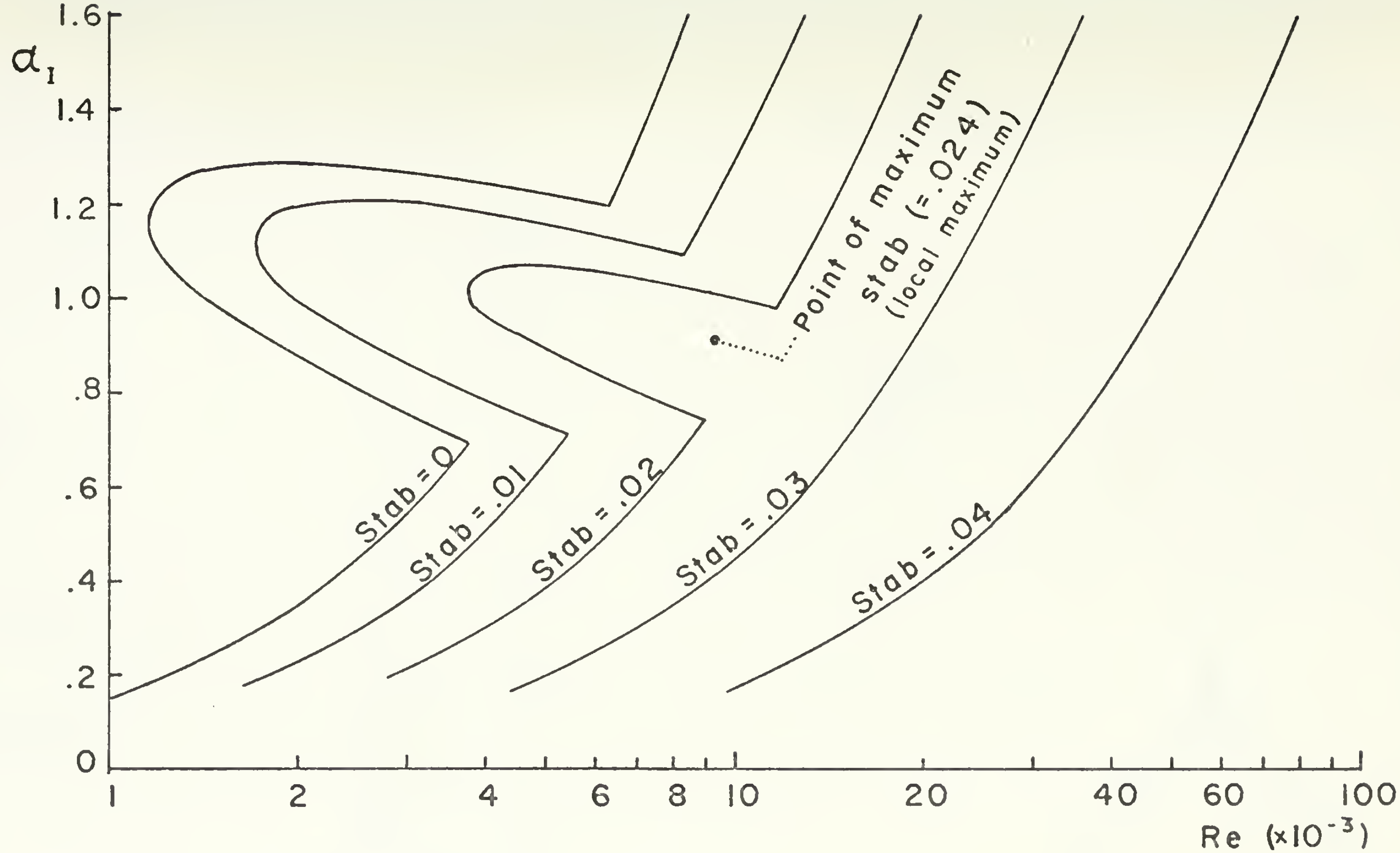

Figure 4 - 5 Curves of constant stability for $\alpha_R = -.04$

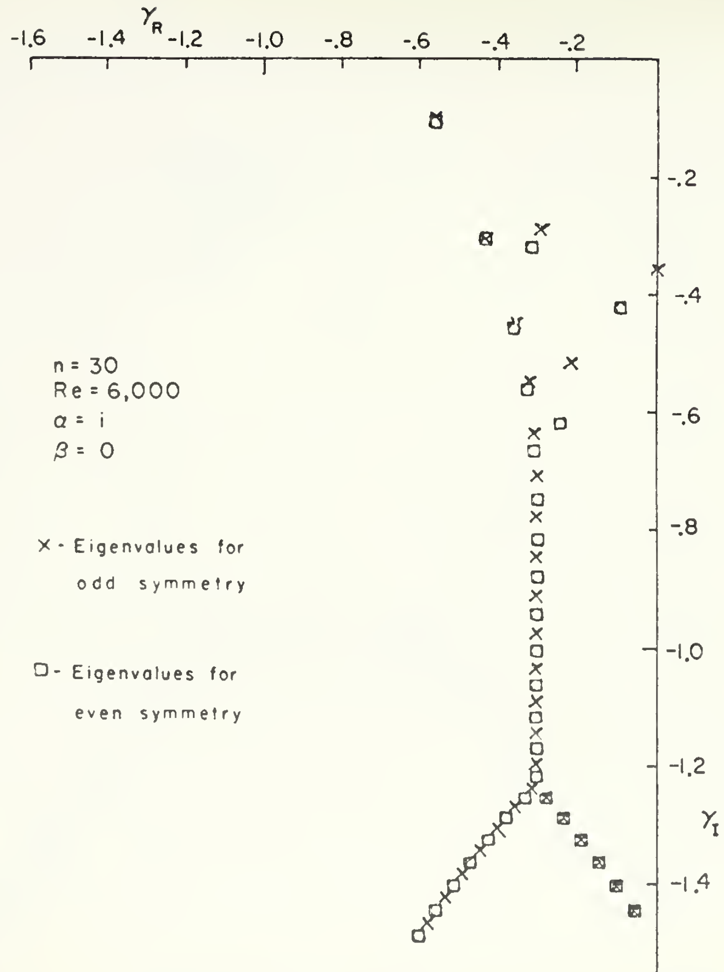

Figure 4-6a n = 30

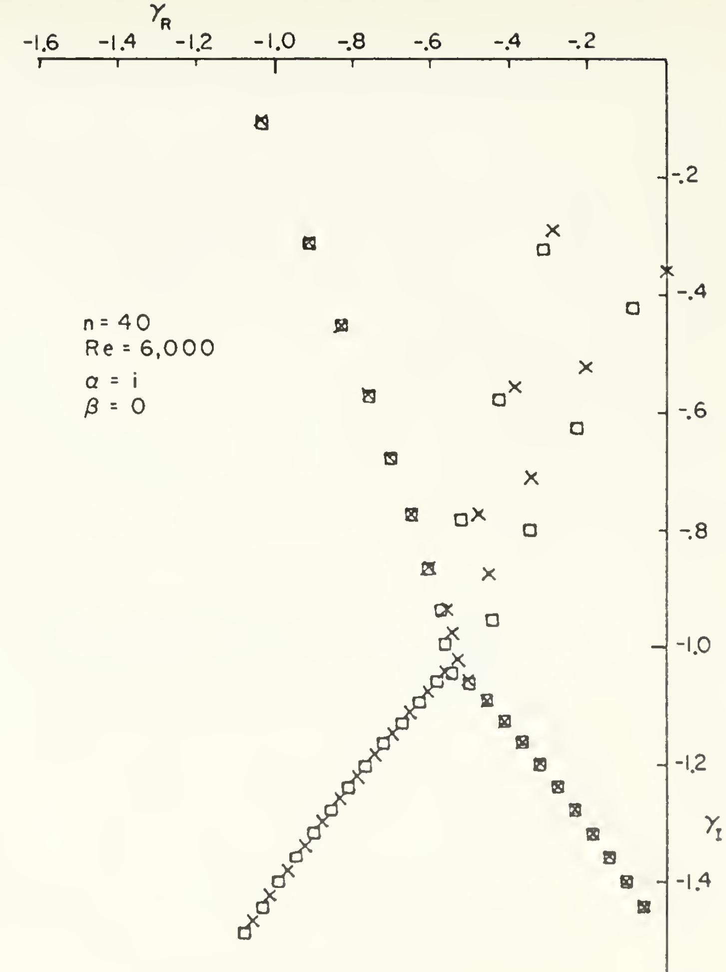

Figure 4-6b n = 40

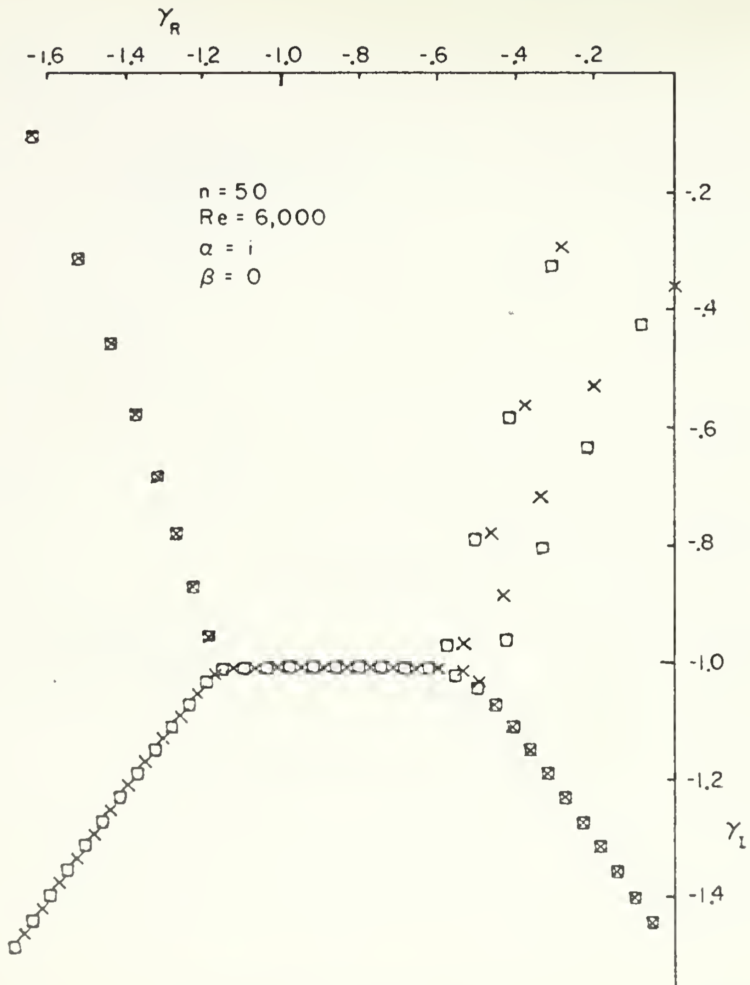

Figure 4-6c n = 50

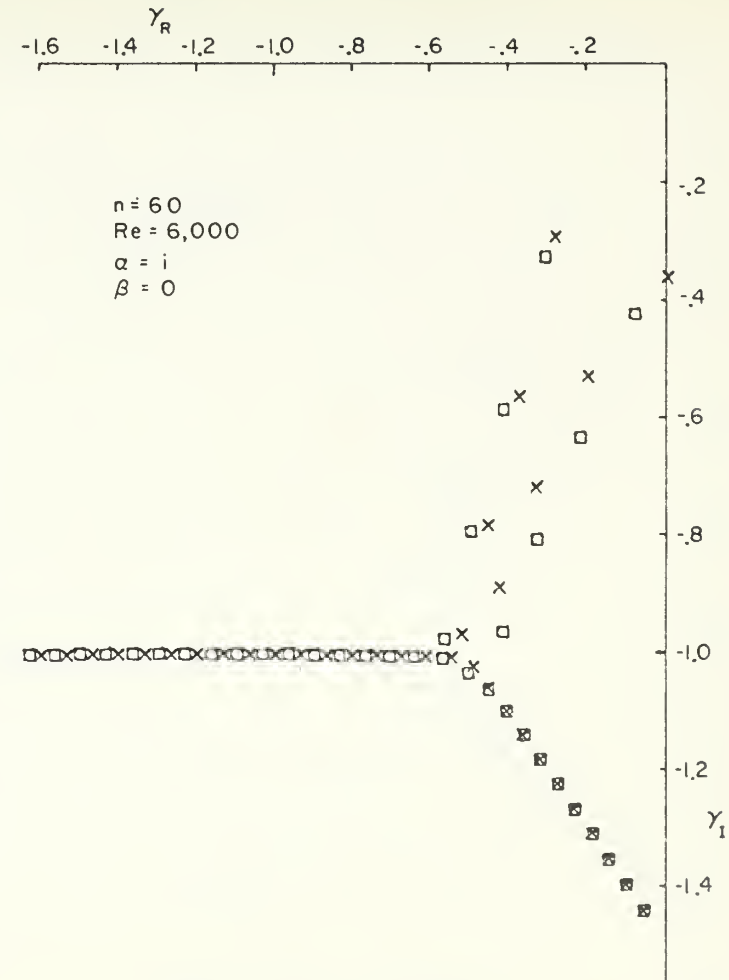

Figure 4-6d n = 60

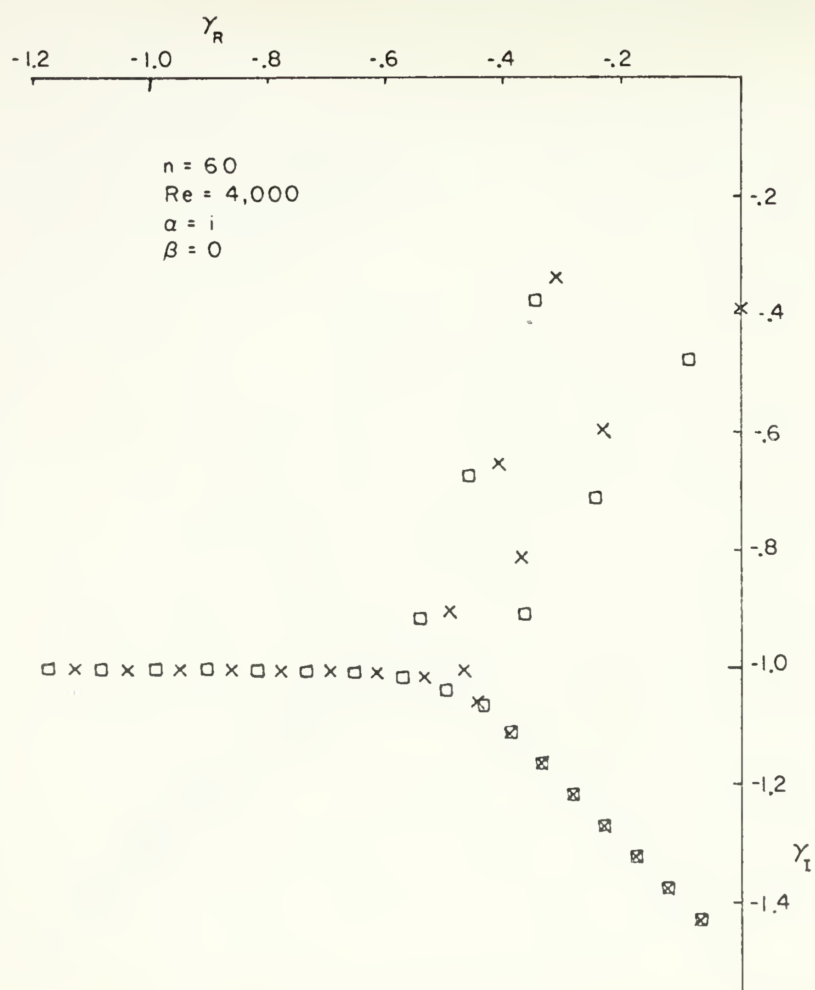

Figure 4-7a Re = 4000

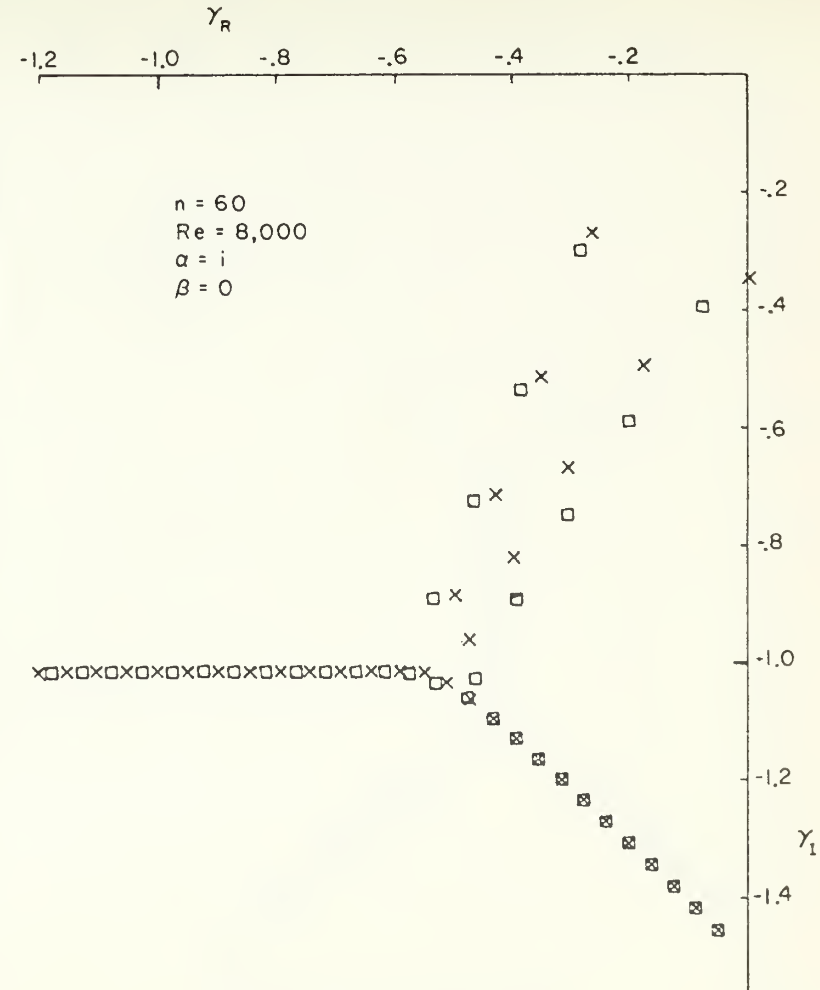

Figure 4-7b Re = 8000

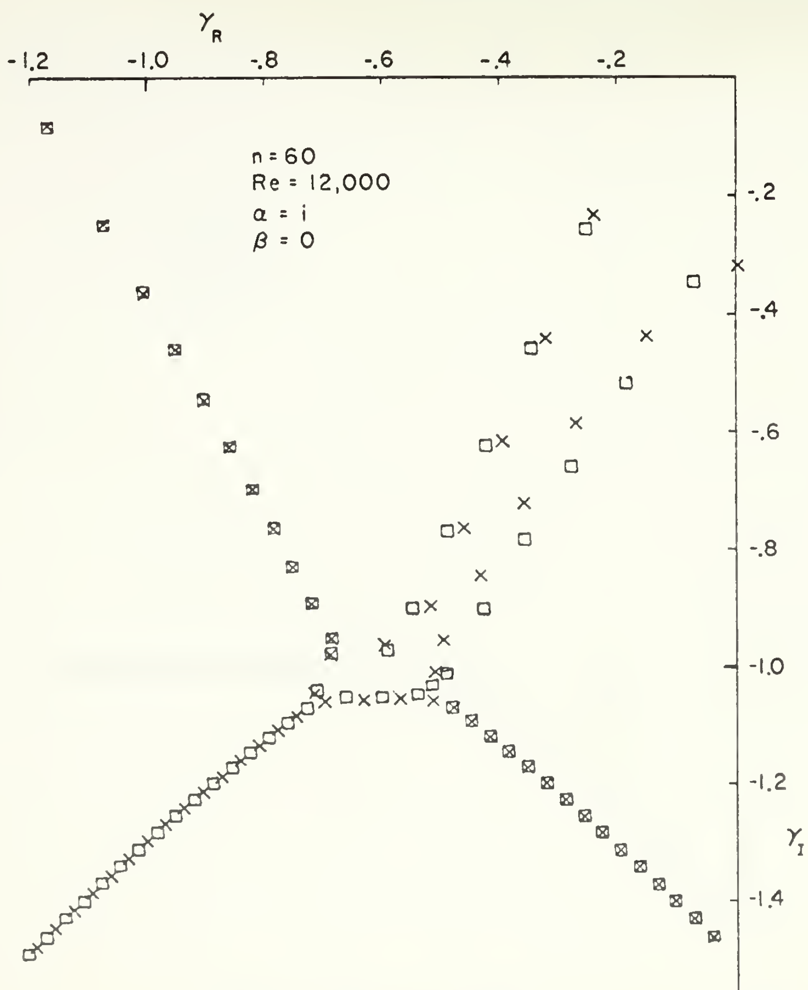

Figure 4-7c Re = 12,000

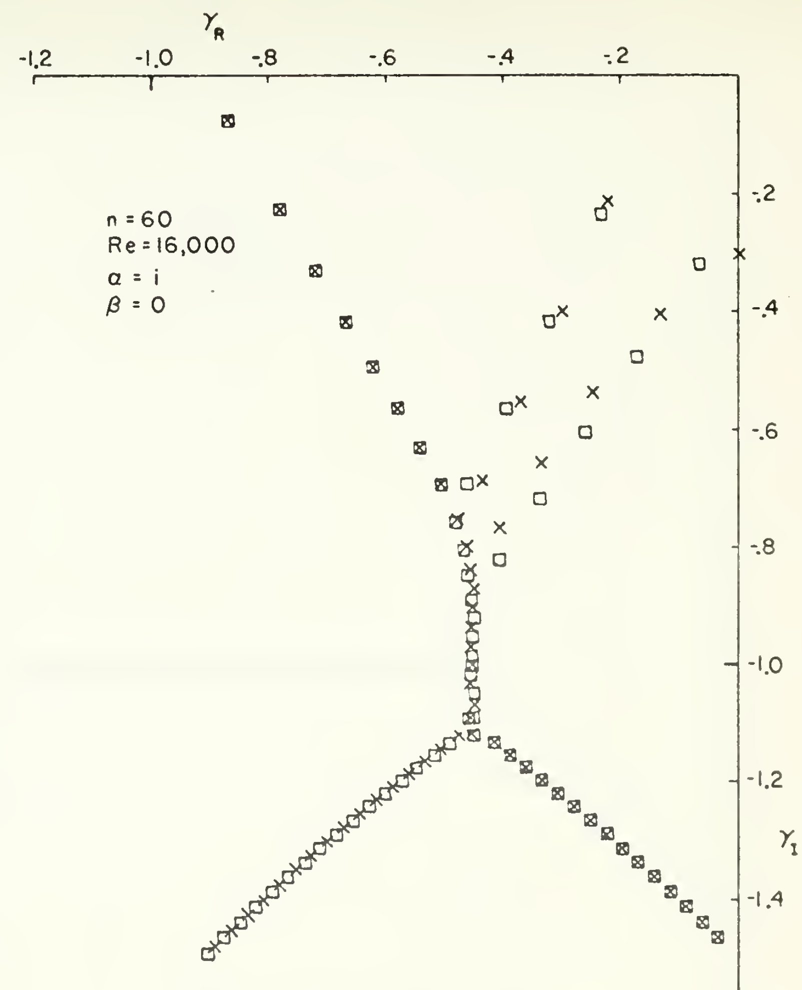

Figure 4-7d Re = 16,000

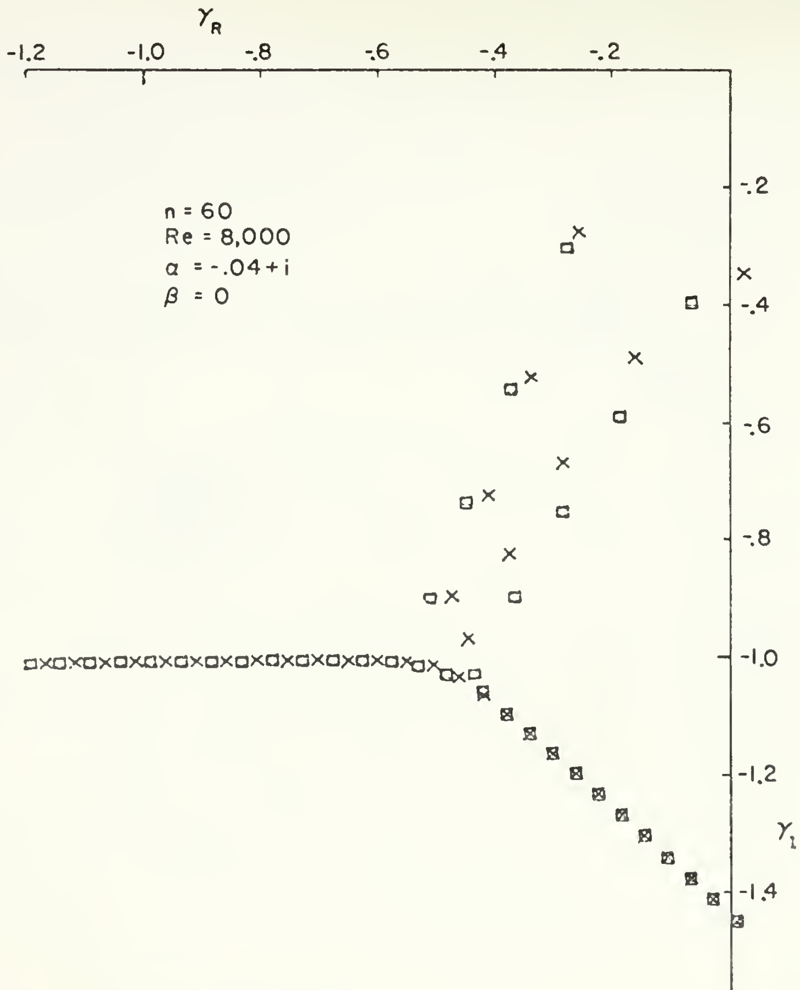

Figure 4-8a α_R = -.04

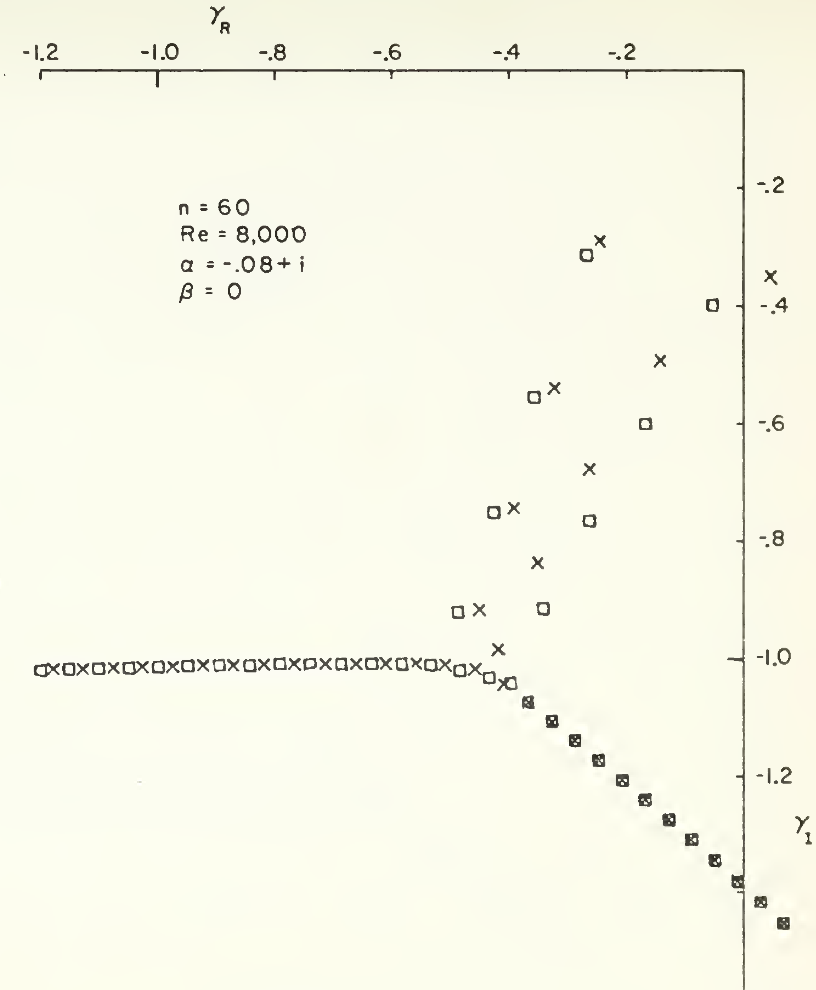

Figure 4-8b α_R = -.08

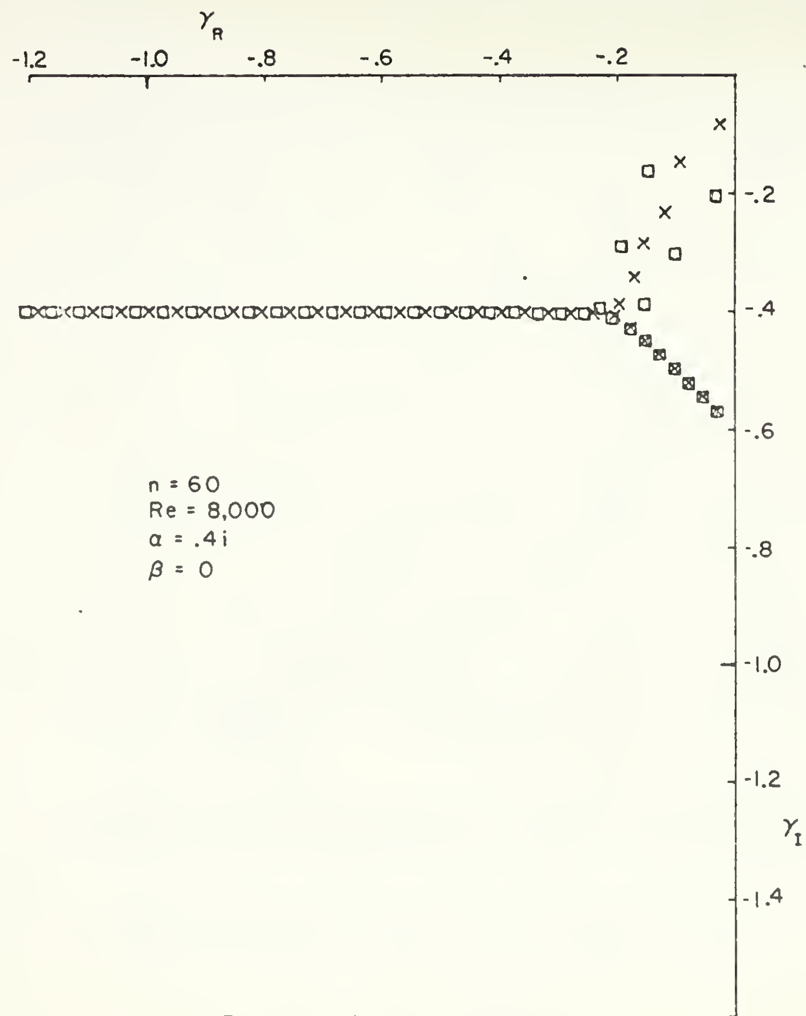

Figure 4-9a $\alpha_I = .4$

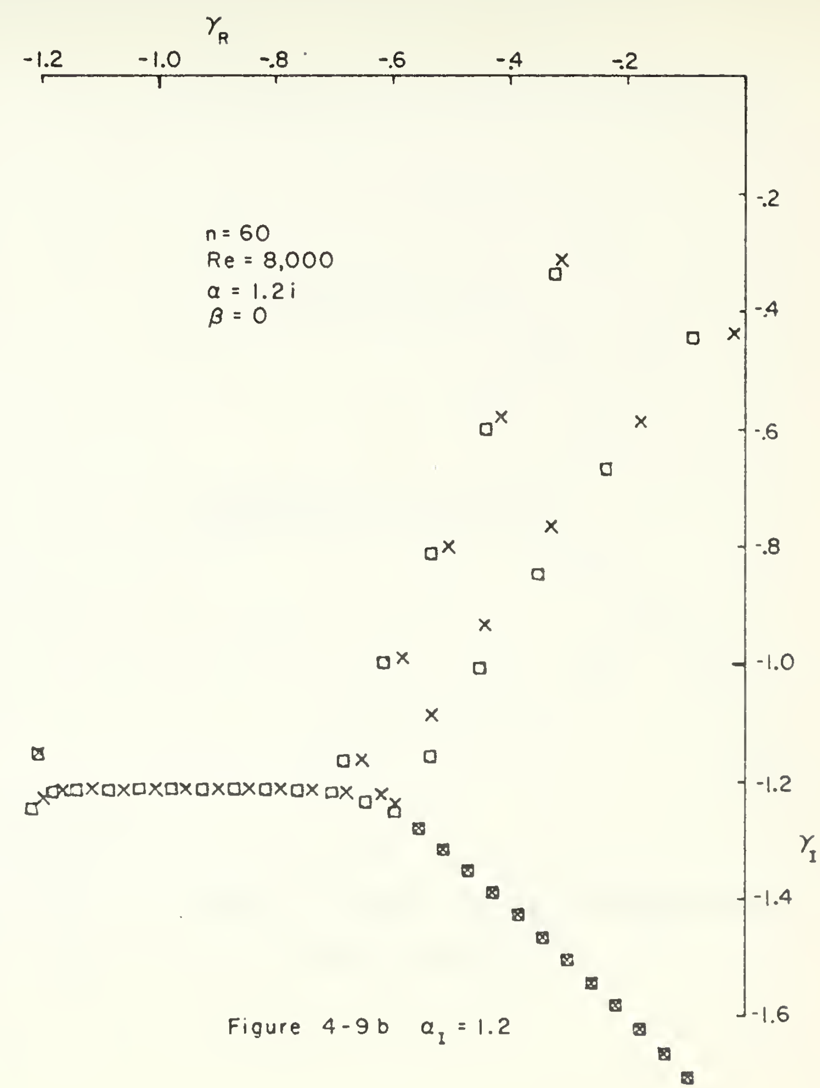

Figure 4-9b $\alpha_I = 1.2$

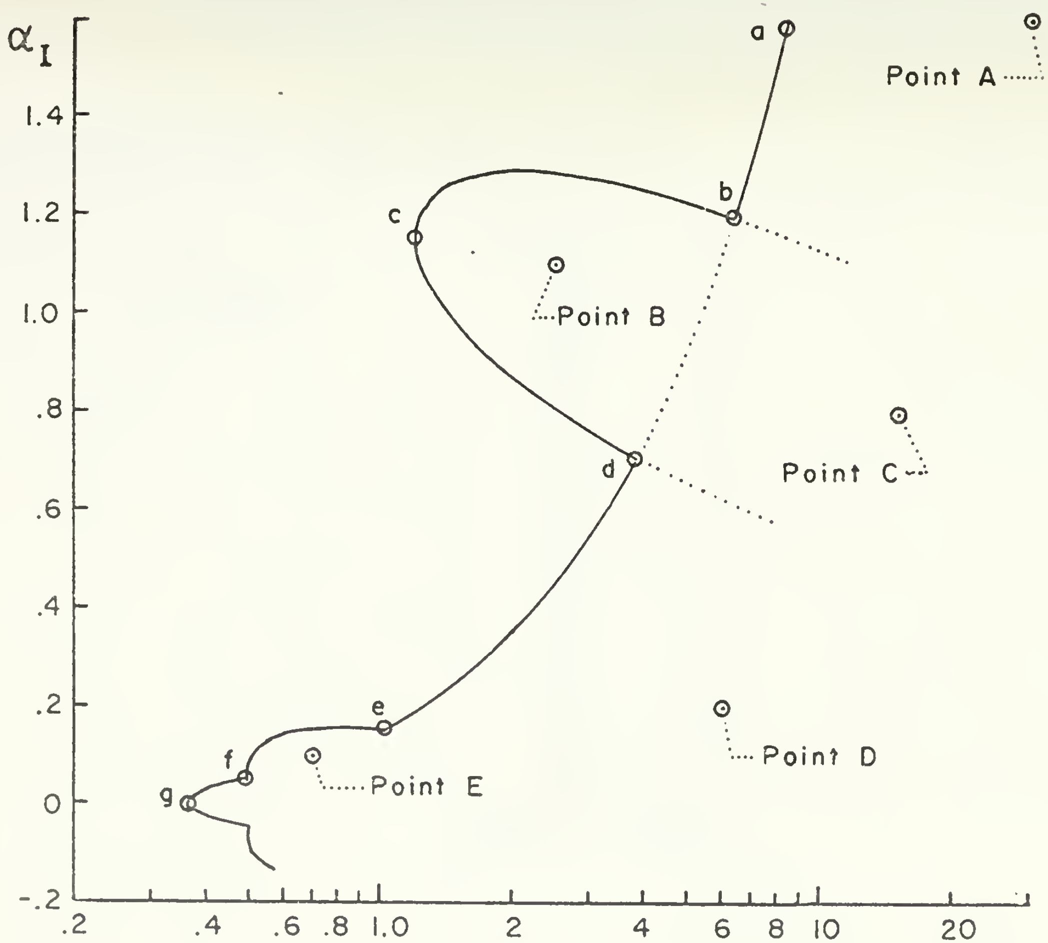

Figure 4-10a Neutral stability curve for $\alpha_R = -.04$, $\beta_I = 0$

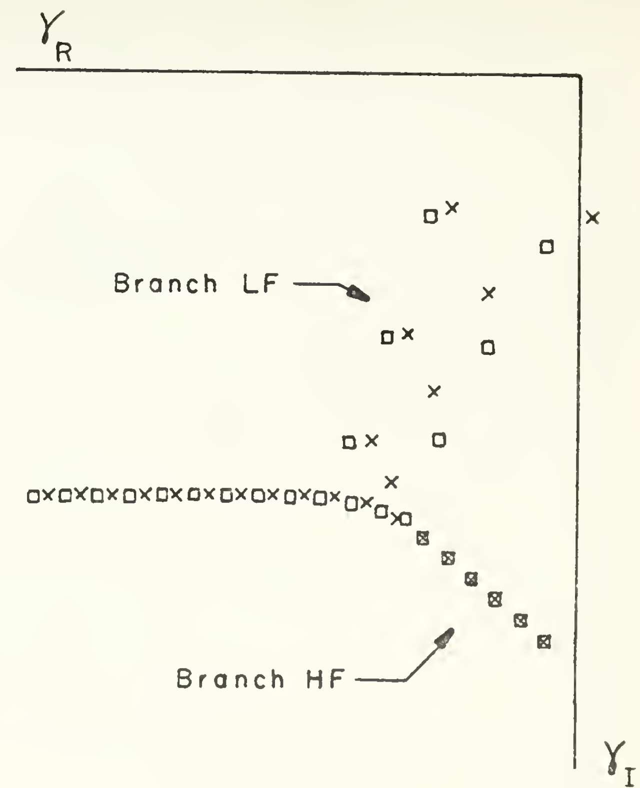

Figure 4-10b Typical eigenvalue plot for large n

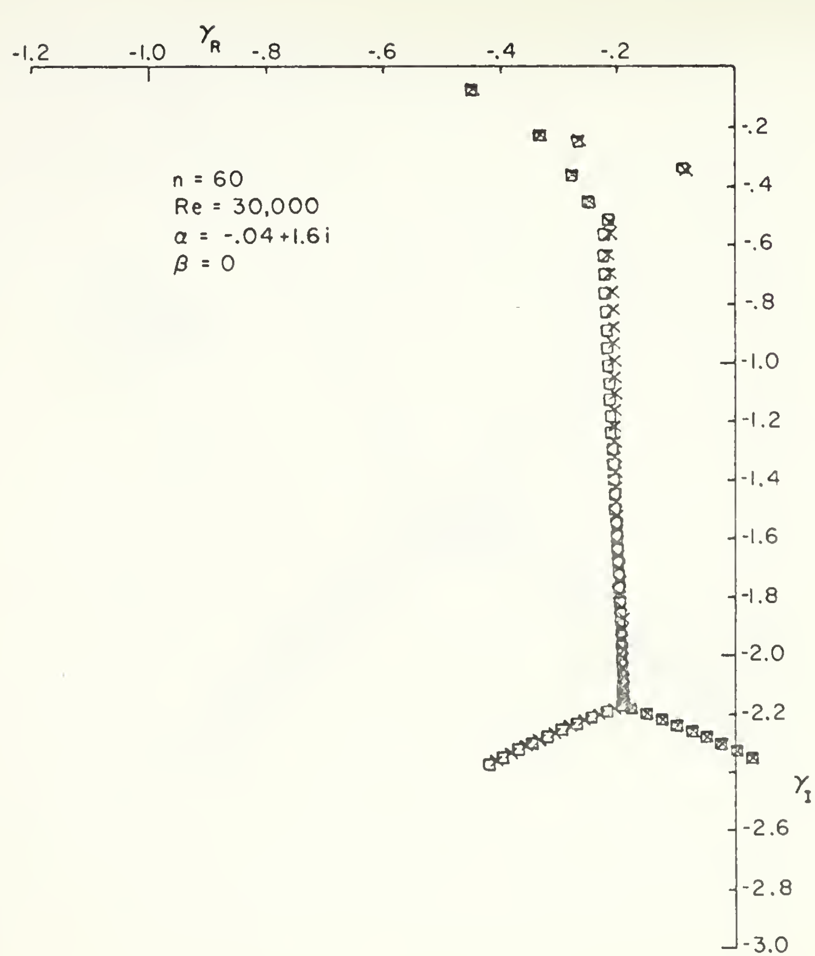

Figure 4-11a Point A

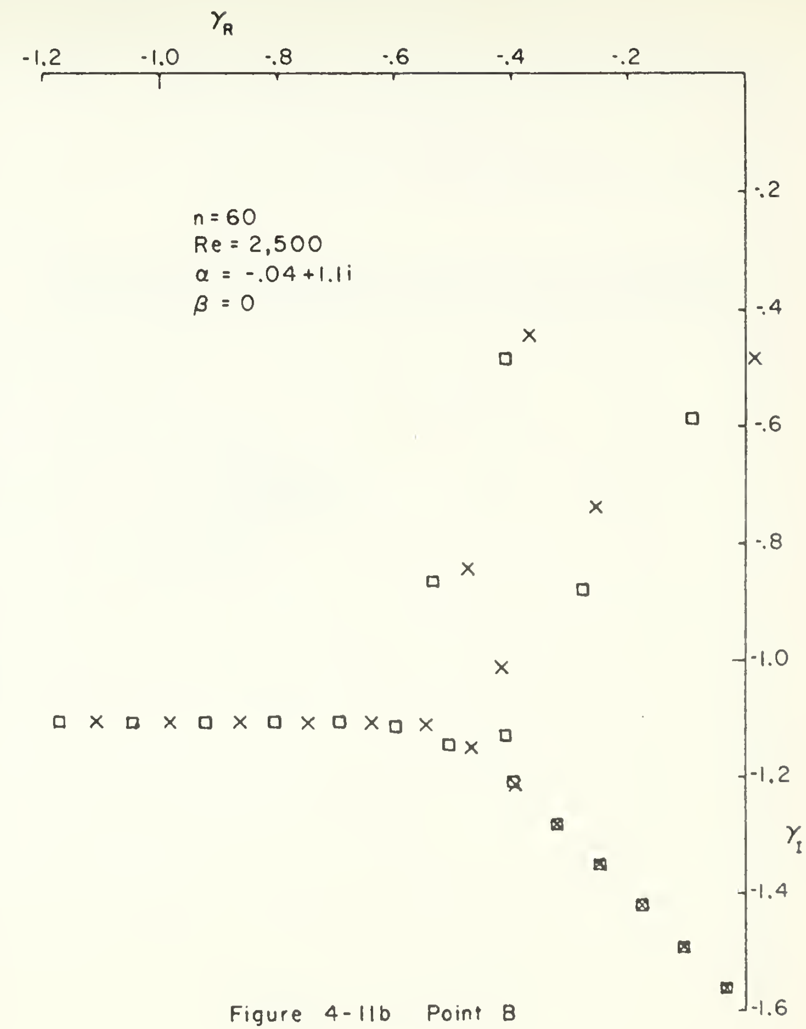

Figure 4-11b Point B

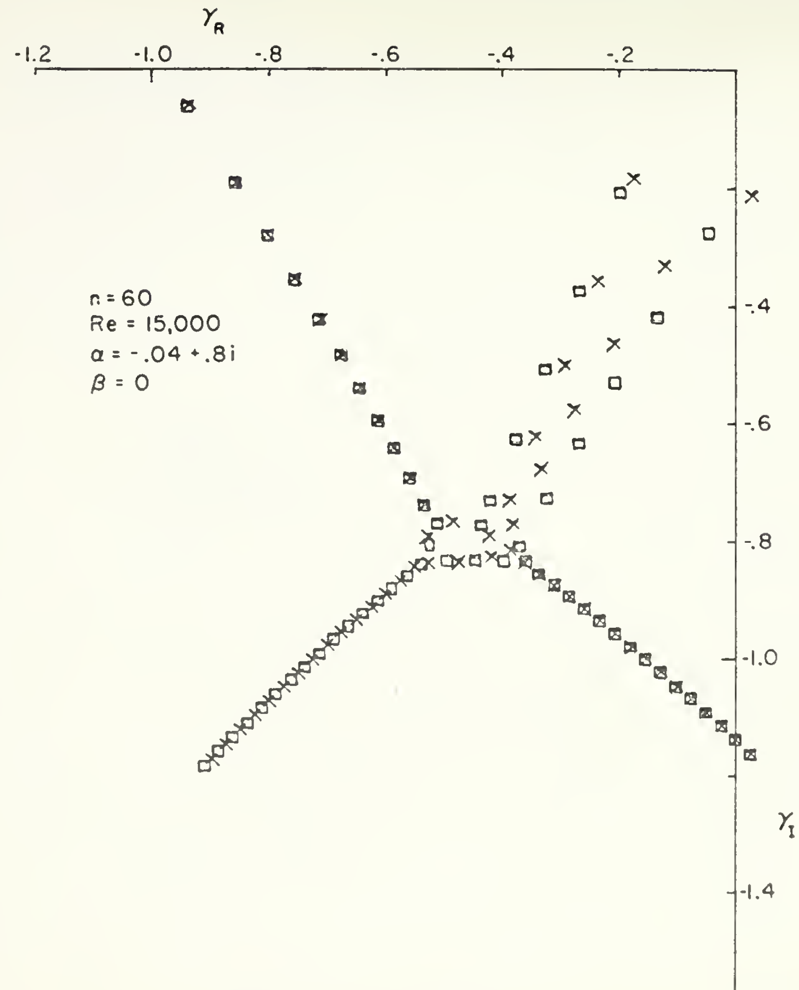

Figure 4-11c Point C

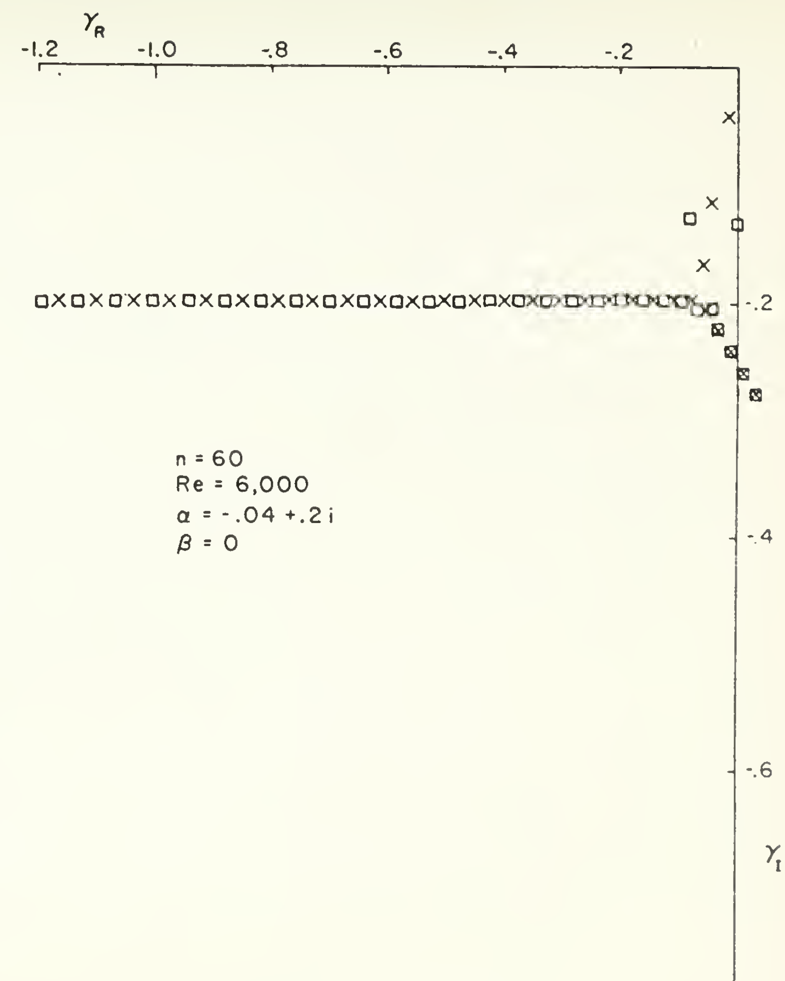

Figure 4-11d Point D

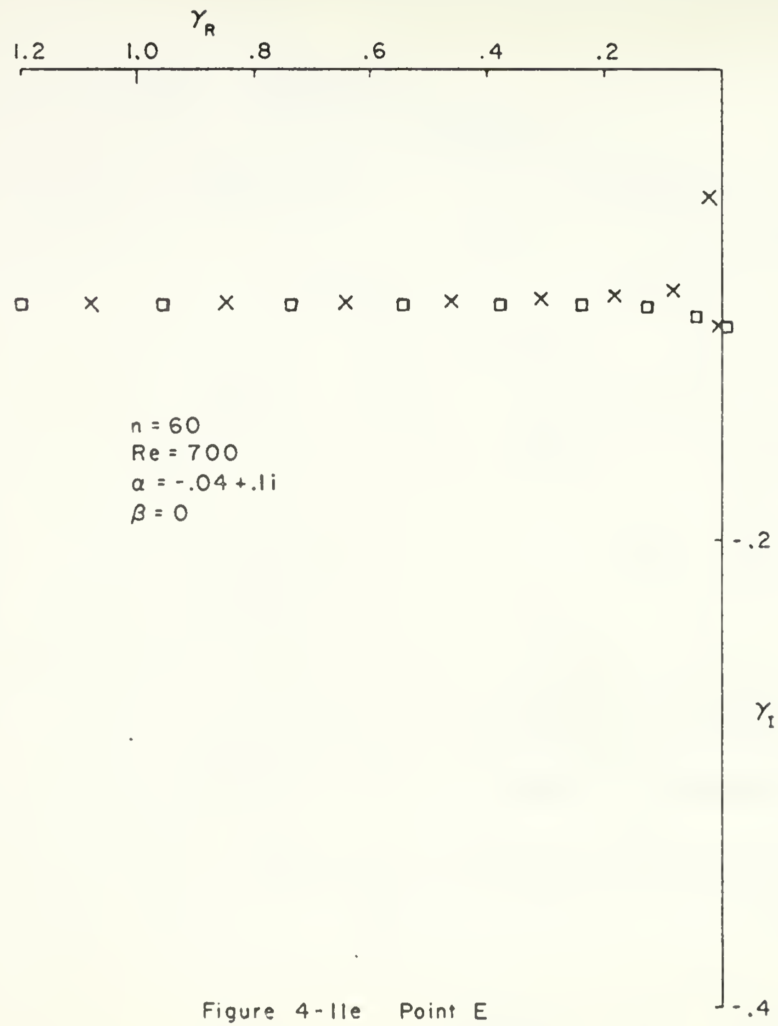

Figure 4-11e Point E

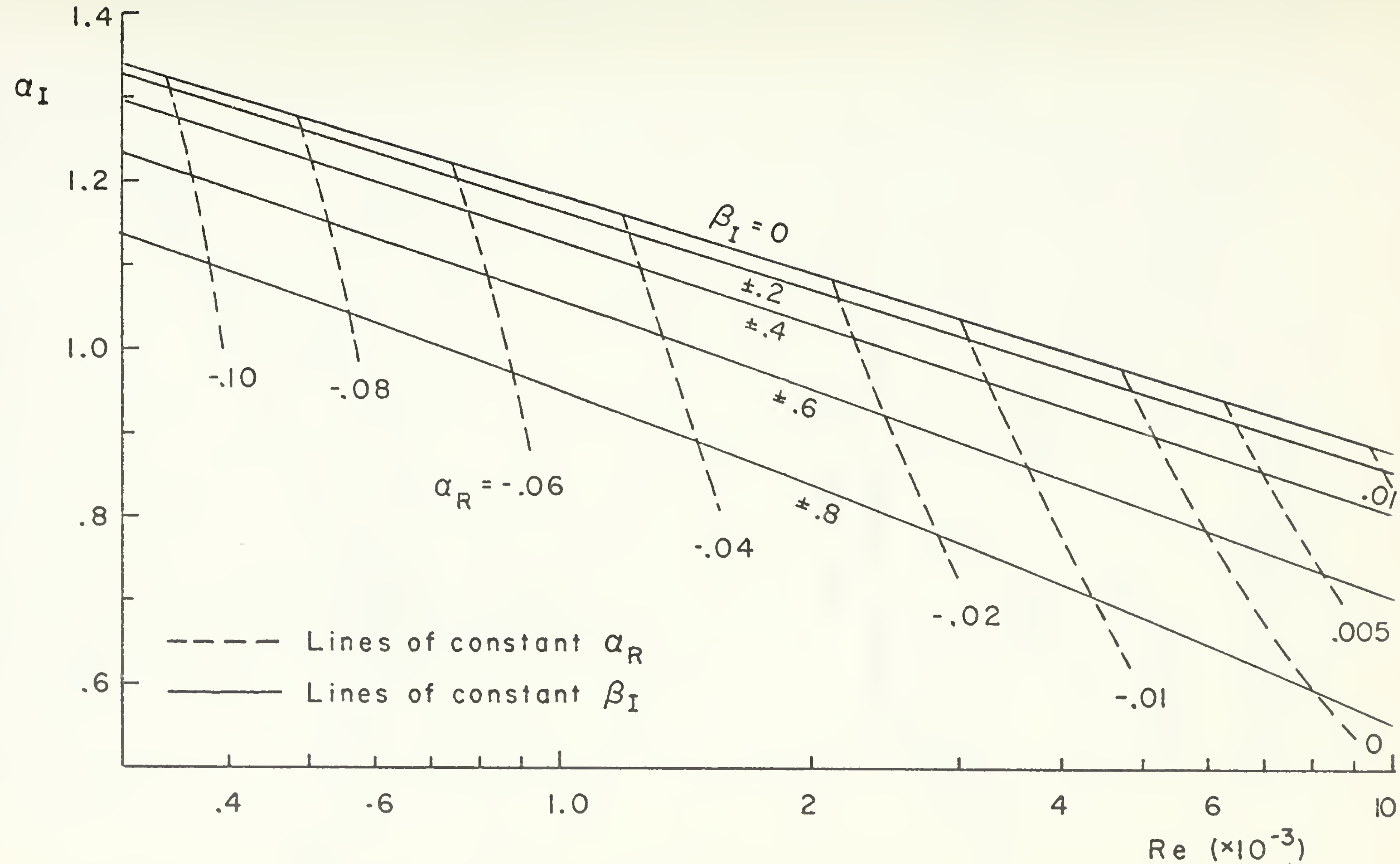

Figure 4-12 Points of minimum Re on neutral stability curves

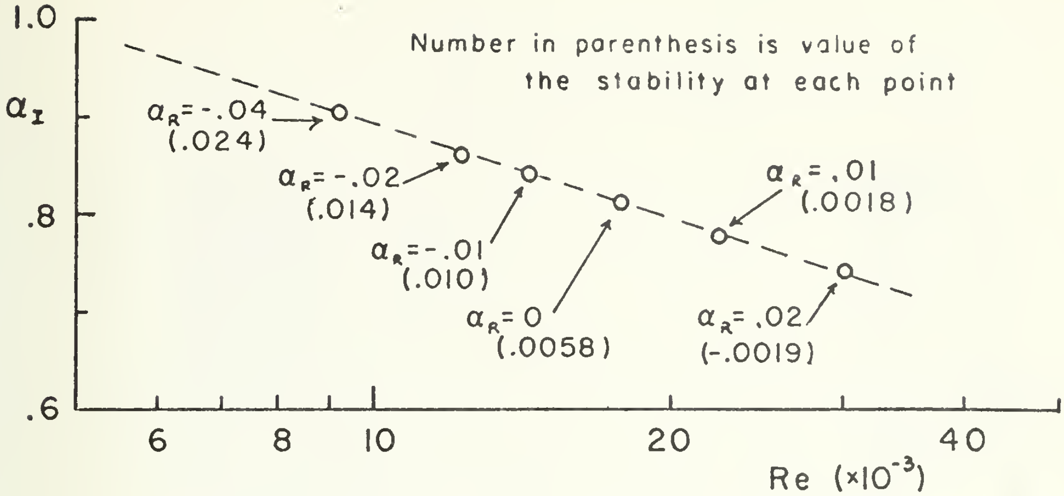

Figure 4-13 Points of maximum instability of upper lobe

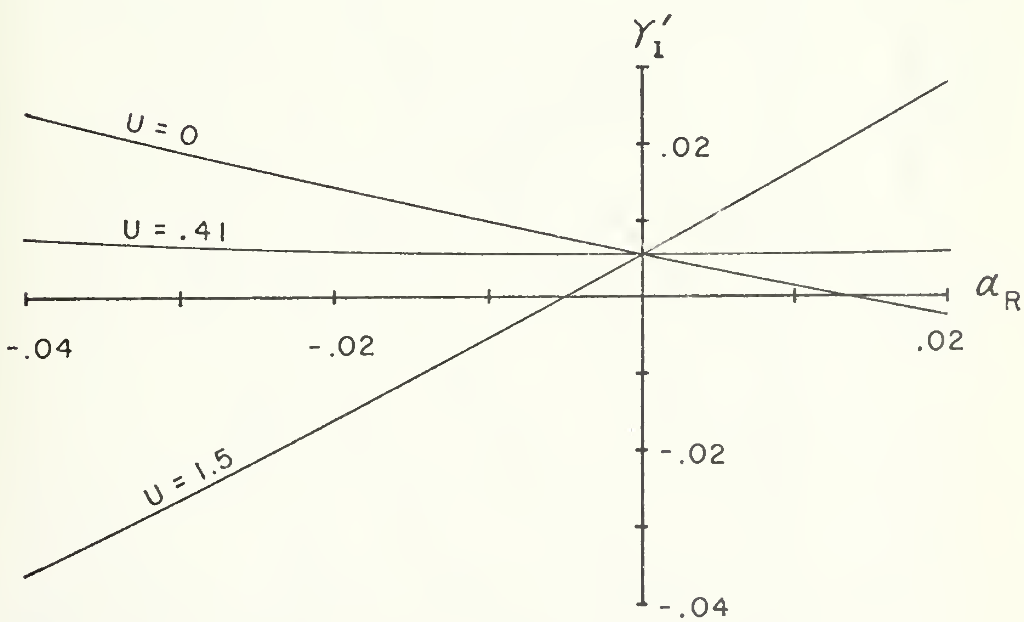

Figure 4-14 Values of maximum instability of the upper lobe vs. α_R for various velocities

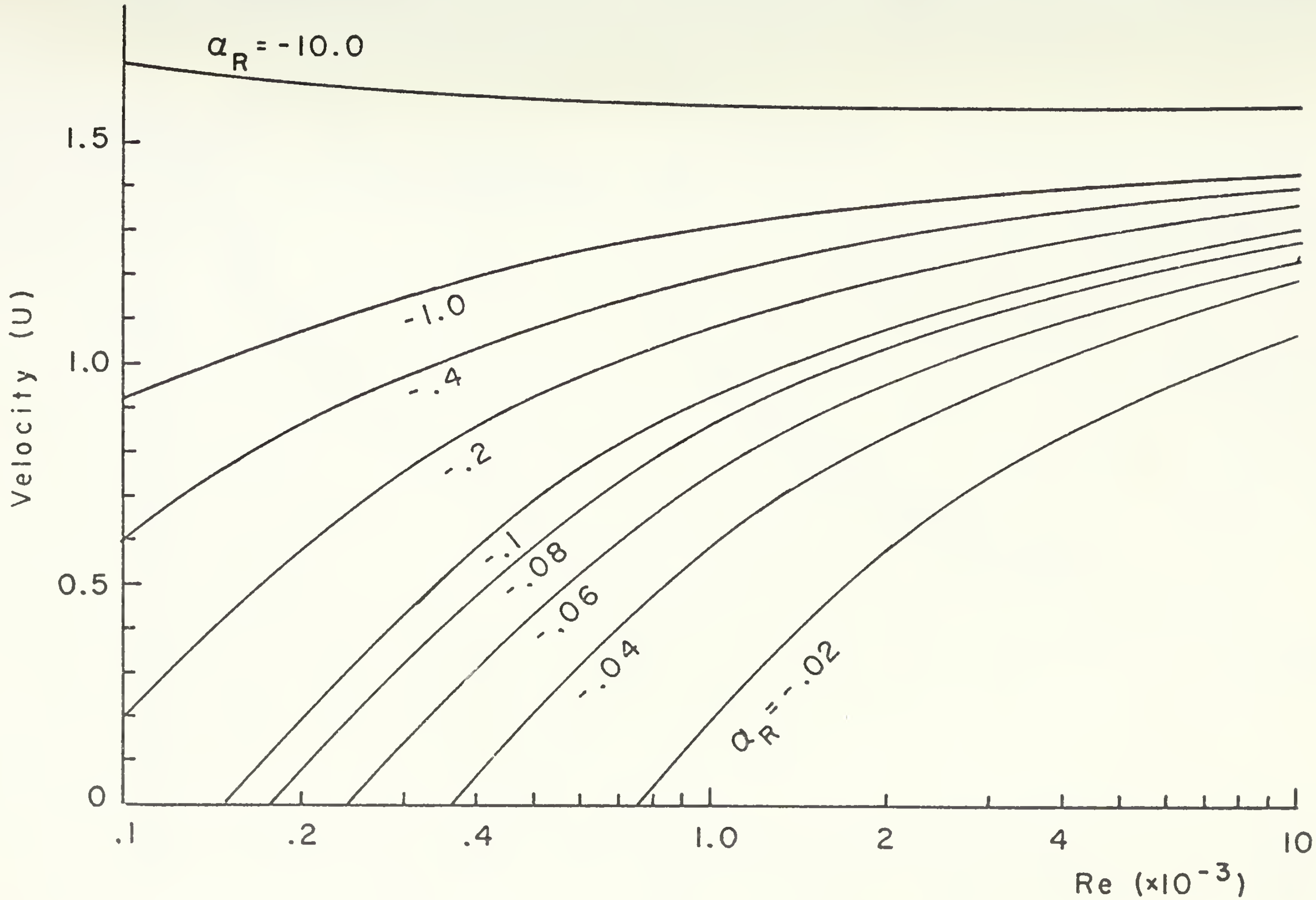

Figure 4-15 Re for neutral stability of the lower lobes for various velocities and α_R

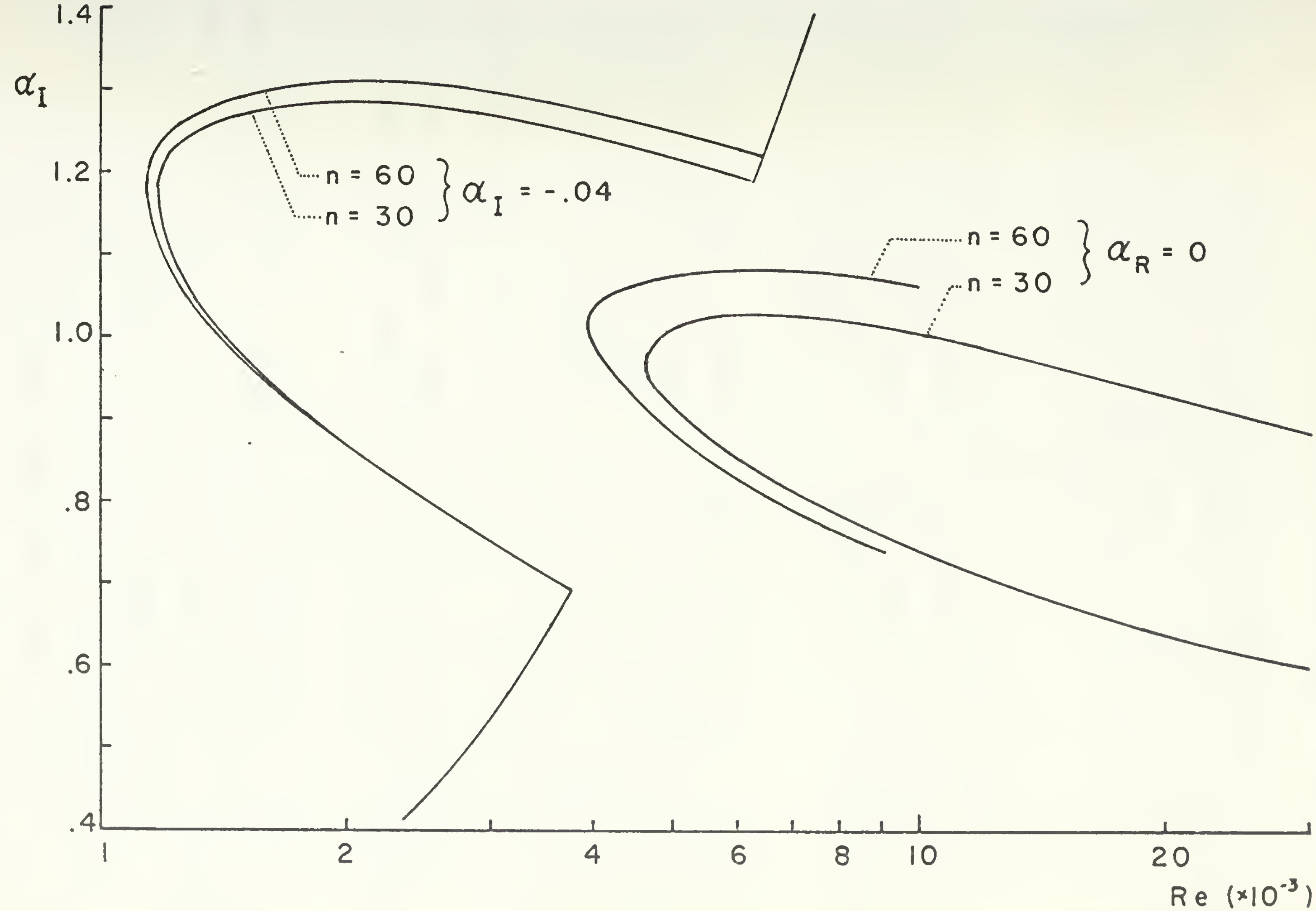

Figure 4-16 Numerical accuracy of two stability curves

V. CONCLUSIONS AND RECOMMENDATIONS

The solution of the linearized vorticity transport equation was simplified considerably by the introduction of the velocity vector potential. This method has not been mentioned in previous publications, at least not in those with which the author is familiar.

The effect of the number of finite difference mesh points on the eigenvalues is important for understanding the effects of the other parameters. This is explained in the results section, and is believed to be information not previously published.

The results presented here show that the critical Reynolds number for plane Poiseuille flow can be lowered as much as desired by proper selection of other parameters. In particular, it is highly significant that a negative value of α_R, that is, streamwise decay in space, produces highly unstable rates of growth in time. The lowering of the critical Reynolds number in this way provides a new basis for improving the agreement between theory and experiment.

Two separate modes of instability were found corresponding to the two branches of the eigenvalue plots. The classical case of instability found for $\alpha_R=0$ corresponds to only one of these branches becoming unstable. It is significant that for each different value of α_R, this mode has a maximum instability associated with it. For a fluid particle velocity of approximately .41, this maximum

instability seems to be invariant with α_R. An additional mode of instability, corresponding to longer wavelengths, appears for nonzero α_R. No maximum value of instability is present as in the first case. This mode has been shown to be unstable at all values of Reynolds number for sufficiently negative values of α_R.

The stability of a fluid particle has been shown to depend upon its velocity. The fact that negative α_R yields the greater instabilities indicates that fluid particles with the lowest velocities, that is, those nearest the walls, will be the most unstable. This seems to agree with experiment. For example, from Schlichting (1968), the transition is "characterized by an amplification of the initial disturbances and by the appearance of self-sustaining turbulent flashes which emanate from fluid layers near the wall along the tube". Further comparison with experiment is recommended.

The eigenvectors of the solutions to the vorticity transport equation were not investigated. Those corresponding to the least stable eigenvalues should be studied.

Since satisfactory results were not obtained for pipe flow, apparently because of an error, an independent study of this case should be made, including the boundary conditions and the numerical methods. It seems likely that the solutions will be qualitatively similar to those obtained for plane flow. If this be true, then

instabilities should be found for negative values of α_R. This would be significant because it is generally agreed that for $\alpha_R=0$, pipe flow is stable to infinitesimal perturbations.

APPENDIX A

DERIVATION OF VORTICITY TRANSPORT EQUATION

The flow of an incompressible fluid with constant viscosity is governed by the Navier-Stokes equation

$$\nu \nabla^2 \bar{V}_d - (\bar{V}_d \cdot \nabla) \bar{V}_d - \nabla p_d / \rho - \partial \bar{V}_d / \partial t_d = 0 \qquad \text{(A-1)}$$

where the subscript d indicates the dimensional quantities of the velocity vector, pressure, and time ($\bar{V}_d$, p_d, and t_d). The kinematic viscosity ν and the density ρ are constants.

This equation is nondimensionalized using a reference length L equal to the half width of the channel in plane flow or the radius of the pipe in pipe flow. The reference velocity is the average velocity of the laminar flow V_{avg}. The resulting equation is

$$\frac{1}{Re} \nabla^2 \bar{V} - (\bar{V} \cdot \nabla) \bar{V} - \nabla p - \partial \bar{V} / \partial t = 0 \qquad \text{(A-2)}$$

where $Re \equiv LV_{avg}/\nu$ is the Reynolds number. This flow is also governed by the continuity equation.

$$\nabla \cdot \bar{V} = 0 \qquad \text{(A-3)}$$

Other than equation (A-1), all the equations presented in this paper are in nondimensional form.

The vorticity transport equation is obtained by taking the curl ($\nabla \times$) of the Navier-Stokes equation. The definition of the vorticity is

$$\bar{\Omega} \equiv \nabla \times \bar{V} \tag{A-4}$$

and the following vector identities are applicable;

$$\nabla \times \nabla p \equiv 0 \tag{A-5a}$$

$$\nabla \times [(\bar{V} \bullet \nabla)\bar{V}] \equiv \nabla \times [\nabla(\bar{V} \bullet \bar{V})/2 - \bar{V} \times (\nabla \times \bar{V})] = -\nabla \times (\bar{V} \times \bar{\Omega}) \tag{A-5b}$$

$$\nabla \times (\nabla^2 \bar{V}) = \nabla^2 \bar{\Omega} \tag{A-5c}$$

$$\nabla \times (\partial \bar{V}/\partial t) = \partial \bar{\Omega}/\partial t \quad . \tag{A-5d}$$

Combining equations (A-4), (A-5), and the curl of (A-2), the result is

$$\frac{1}{Re}\nabla^2 \bar{\Omega} + \nabla \times (\bar{V} \times \bar{\Omega}) - \partial \bar{\Omega}/\partial t = 0 \ . \tag{A-6}$$

Next, it is assumed that the flow is made up of the steady-state, laminar solution for a given flow geometry and a small perturbation flow which is added to the laminar flow. The velocity and vorticity become, respectively, $\bar{V}+\bar{v}$ and $\bar{\Omega}+\bar{\omega}$ where $\bar{v}$ and $\bar{\omega}$ are the perturbation quantities. It is easy to show that the perturbation flow must satisfy continuity independently.

$$\nabla \bullet \bar{v} = 0 \tag{A-7}$$

It can also be shown that

$$\bar{\omega} = \nabla \times \bar{v} \ . \tag{A-8}$$

Substituting $\bar{V}+\bar{v}$ and $\bar{\Omega}+\bar{\omega}$ into the vorticity transport

equation and subtracting the equation for the unperturbed flow, the result is

$$\frac{1}{Re}\nabla^2\bar{\omega} + \nabla\times(\bar{V}\times\bar{\omega}-\bar{\Omega}\times\bar{v}+\bar{v}\times\bar{\omega}) - \frac{\partial\bar{\omega}}{\partial t} = 0 \,. \tag{A-9}$$

This equation is linear in the perturbation quantities except for the second order term $\bar{v}\times\bar{\omega}$. If the perturbation is small, this term can be neglected, linearizing the equation. The result is the linearized perturbation vorticity transport equation.

$$\frac{1}{Re}\nabla^2\bar{\omega} + \nabla\times(\bar{V}\times\bar{\omega}-\bar{\Omega}\times\bar{v}) - \frac{\partial\bar{\omega}}{\partial t} = 0 \tag{A-10}$$

A second form of this equation is derived by expanding the curl of the cross product terms and noting that $\nabla\bullet\bar{V} = \nabla\bullet\bar{v} = \nabla\bullet\bar{\Omega} = \nabla\bullet\bar{\omega} = 0$. The result is

$$(1/Re)\,\nabla^2\bar{\omega} + (\bar{\omega}\bullet\nabla)\,\bar{V} - (\bar{V}\bullet\nabla)\,\bar{\omega} + (\bar{v}\bullet\nabla)\,\bar{\Omega}$$
$$- (\bar{\Omega}\bullet\nabla)\,\bar{v} - \partial\bar{\omega}/\partial t = 0 \,. \tag{A-11}$$

A third form of equation (A-10) is obtained using the vector identity

$$\nabla^2\bar{\omega} = \nabla(\nabla\bullet\bar{\omega}) - \nabla\times(\nabla\times\bar{\omega}) \tag{A-12}$$

and noting that $\bar{\omega}$ is non-divergent.

$$-\frac{1}{Re}\nabla\times\nabla\times\bar{\omega} + \nabla\times\bar{V}\times\bar{\omega} - \nabla\times\bar{\Omega}\times\bar{v} - \frac{\partial\bar{\omega}}{\partial t} = 0 \tag{A-13}$$

APPENDIX B

DEPENDENCE BETWEEN COMPONENTS OF THE PERTURBATION VORTICITY TRANSPORT EQUATION

The vorticity transport equation

$$-\frac{1}{Re}\nabla\times\nabla\times\bar{\omega} + \nabla\times\bar{V}\times\bar{\omega} - \nabla\times\bar{\Omega}\times\bar{v} - \frac{\partial\bar{\omega}}{\partial t} = 0 \qquad \text{(A-13)}$$

appears superficially to represent a set of three, coupled, simultaneous equations. This equation actually has only two independent conditions. To show this, equation (A-13) is first simplified by the introduction of the velocity vector potential $\bar{W}$ defined by

$$\bar{v} = \nabla\times\bar{W} \ . \qquad \text{(2-7)}$$

In Appendix D it is shown that this results in a vector equation which contains only the three components of $\bar{W}$ as its unknowns.

$$-\frac{1}{Re}\nabla\times\nabla\times\nabla\times\nabla\times\bar{W} + \nabla\times\bar{V}\times\nabla\times\nabla\times\bar{W} - \nabla\times\bar{\Omega}\times\nabla\times\bar{W}$$

$$- \nabla\times\nabla\times\frac{\partial\bar{W}}{\partial t} = 0 \qquad \text{(D-1)}$$

Furthermore, the following form of solution is assumed.

$$\bar{W}(x,y,z,t) = [\bar{i}f(y) + \bar{j}g(y) + \bar{k}h(y)]e^{X} \qquad \text{(2-9a)}$$

where

$$X \equiv \alpha x + \beta z + \gamma t \ . \qquad \text{(2-10)}$$

Consider equations (D-1) expressed in abbreviated form by the statement

$$\bar{\Gamma} = \begin{Bmatrix} \Gamma_x \\ \Gamma_y \\ \Gamma_z \end{Bmatrix} = 0 \tag{B-1}$$

which may be expressed as three separate equations.

$$\Gamma_x = 0 \tag{B-2a}$$

$$\Gamma_y = 0 \tag{B-2b}$$

$$\Gamma_z = 0 \tag{B-2c}$$

Equations (B-2) are written for cartesian coordinates. The components Γ_x, Γ_y and Γ_z happen to be the three components of a vector obtained by applying the curl operator to the momentum transport equation. It is a vector identity that the curl of any vector is nondivergent and thus

$$\frac{\partial}{\partial x}(\Gamma_x) + \frac{\partial}{\partial y}(\Gamma_y) + \frac{\partial}{\partial z}(\Gamma_z) = 0 \tag{B-3}$$

With the assumed form of $\bar{W}$ given by equations (2-9a) and (2-10) and the fact that all these equations are linear in $\bar{W}$, (B-3) becomes

$$\alpha\Gamma_x + \frac{\partial}{\partial y}(\Gamma_y) + \beta\Gamma_z = 0 \tag{B-4}$$

Equation (B-4) has been verified for the resulting form of the equation developed in Appendix D (equations (D-46) through (D-55)) by differentiating the second equation, multiplying the first by α and the third by β, and adding the three equations thus obtained. All terms are found to cancel out. From equation (B-4) it may be inferred that it

is sufficient to satisfy the second of equations (B-2) and any linear combination of the first and third. If the second is satisfied, then

$$\Gamma_y = 0 \tag{B-5}$$

and it must be true that

$$\frac{\partial}{\partial y}(\Gamma_y) = 0 \; . \tag{B-6}$$

Equation (B-4) reduces to

$$\alpha\Gamma_x + \beta\Gamma_z = 0 \; . \tag{B-7}$$

As discussed in section 2, the proper choice of a second condition is

$$-\beta\Gamma_x + \alpha\Gamma_z = 0 \tag{2-11}$$

Equations (B-7) and (2-11) may be expressed in matrix form as

$$\begin{bmatrix} \alpha & \beta \\ -\beta & \alpha \end{bmatrix} \begin{Bmatrix} \Gamma_x \\ \Gamma_z \end{Bmatrix} = \begin{Bmatrix} 0 \\ 0 \end{Bmatrix} \tag{B-8}$$

If the determinant is non-vanishing, that is, if

$$\alpha^2 + \beta^2 \neq 0 \; , \tag{B-9}$$

then the only solution of equation (B-8) is

$$\begin{Bmatrix} \Gamma_x \\ \Gamma_z \end{Bmatrix} = \begin{Bmatrix} 0 \\ 0 \end{Bmatrix} . \tag{B-10}$$

Thus, equations (B-5) and (B-10) verify that equation (B-1) will be satisfied by the two equations (B-5) and (2-11). That is, these two conditions are sufficient.

For pipe flow in cylindrical coordinates, equation (D-1)

may be expressed similarly to (B-1) as

$$\bar{\Gamma} = \begin{Bmatrix} \Gamma_x \\ \Gamma_r \\ \Gamma_\theta \end{Bmatrix} = 0 \qquad \text{(B-11)}$$

which are three separate equations.

$$\Gamma_x = 0 \qquad \text{(B-12a)}$$

$$\Gamma_r = 0 \qquad \text{(B-12b)}$$

$$\Gamma_\theta = 0 \qquad \text{(B-12c)}$$

As in the cartesian case, equation (B-11) must be nondivergent.

$$\frac{\partial}{\partial x}(\Gamma_x) + \frac{1}{r}\frac{\partial}{\partial r}(r\Gamma_r) + \frac{1}{r}\frac{\partial}{\partial \theta}(\Gamma_\theta) = 0 \qquad \text{(B-13)}$$

With the assumed form of $\bar{W}$ given in equations (E-1) and (E-2), equation (B-13) becomes

$$\alpha\Gamma_x + \frac{1}{r}D(r\Gamma_r) + \frac{\beta}{r}\Gamma_\theta = 0 \qquad \text{(B-14)}$$

Equation (B-14) has been verified by performing the indicated operations on Γ_x, Γ_r, and Γ_θ. All terms cancel out.

From equation (E-14) it may be inferred that it is sufficient to satisfy equation (B-12b) and any linear combination of Γ_x and Γ_θ. If the second is satisfied then

$$\Gamma_y = 0 \ . \qquad \text{(B-15)}$$

Therefore

$$\frac{1}{r}D(r\,\Gamma_r) = 0 \qquad \text{(B-16)}$$

for any value of r except at r=0, where it is true in the limit as r approaches zero. Equation (B-14) then reduces to

$$\alpha\Gamma_x + \frac{\beta}{r}\Gamma_\theta = 0 \qquad \text{(B-17)}$$

As discussed in section 2, the proper choice of a second condition is

$$-\frac{\beta}{r}\Gamma_x + \alpha\Gamma_\theta = 0 \qquad \text{(2-14)}$$

These two equations may be expressed in matrix form as

$$\begin{bmatrix} \alpha & \frac{\beta}{r} \\ -\frac{\beta}{r} & \alpha \end{bmatrix} \begin{Bmatrix} \Gamma_x \\ \Gamma_\theta \end{Bmatrix} = \begin{Bmatrix} 0 \\ 0 \end{Bmatrix} \qquad \text{(B-18)}$$

and, as before, equation (B-11) will be satisfied under the condition that

$$\alpha^2 + \frac{\beta^2}{r^2} \neq 0 \; . \qquad \text{(B-19)}$$

APPENDIX C

REDUNDANCY OF THE THREE COMPONENTS OF THE VELOCITY VECTOR POTENTIAL AND THEIR RELATIONSHIP TO THE STREAMFUNCTION

1. Cartesian Coordinates

To find the relationship between the components of the velocity vector potential in cartesian coordinates, consider the definition of $\bar{W}$.

$$\bar{v} = \nabla \times \bar{W} = \begin{vmatrix} \bar{i} & \bar{j} & \bar{k} \\ \frac{\partial}{\partial x} & \frac{\partial}{\partial y} & \frac{\partial}{\partial z} \\ f(y) & g(y) & h(y) \end{vmatrix} e^{X} = \begin{vmatrix} \bar{i} & \bar{j} & \bar{k} \\ \alpha & D & \beta \\ f & g & h \end{vmatrix} e^{X} \qquad \text{(C-1)}$$

which has the components

$$u(y) = Dh - \beta g \qquad \text{(C-2a)}$$

$$v(y) = \beta f - \alpha h \qquad \text{(C-2b)}$$

$$w(y) = \alpha g - Df \;. \qquad \text{(C-2c)}$$

The components of the velocity, u, v, and w, are not independent but are related by the continuity equation, that is, by the condition of non-divergence.

$$\nabla \bullet \bar{u} = \alpha u + Dv + \beta w = 0 \qquad \text{(C-3)}$$

Let f(y) be identically zero. Then

$$u(y) = Dh - \beta g \qquad \text{(C-4a)}$$

$$v(y) = -\alpha h \qquad \text{(C-4b)}$$

$$w(y) = \alpha g \ . \tag{C-4c}$$

Thus the components of the velocity may be represented by the two components, h and g, of the velocity vector potential without any loss of generality other than that imposed by the condition of non-divergence. If g(y) is set to zero, the following set of equations result.

$$u(y) = Dh \tag{C-5a}$$

$$v(y) = \beta f - \alpha h \tag{C-5b}$$

$$w(y) = -Df \tag{C-5c}$$

And if h(y) is identically zero

$$u(y) = -\beta g \tag{C-6a}$$

$$v(y) = \beta f \tag{C-6b}$$

$$w(y) = \alpha g - Df \ . \tag{C-6c}$$

In the classical case of plane perturbations the parameter β is zero. Then the 3-dimensional equations, (C-2), reduce to

$$u(y) = Dh \tag{C-7a}$$

$$v(y) = -\alpha h \tag{C-7b}$$

$$w(y) = \alpha g - Df \ . \tag{C-7c}$$

From equations (C-7) it can be seen that, for the case of $\beta=0$, letting $h(y)=0$ cannot be allowed because both u(y) and v(y) are forced to be zero. Since it is desirable to have the three-dimensional representation of the problem reducible to the classical two-dimensional case, it is stipulated that the function to be eliminated not be h(y).

Note that the classical case of two-dimensional plane perturbations corresponds, in the present notation, to setting f(y) and g(y) equal to zero and retaining h(y) alone. This component of the velocity vector potential corresponds to the classical definition of the stream function for two-dimensional flow. From equations (C-7)

$$u(y) = Dh \tag{C-8a}$$

$$v(y) = -\alpha h \tag{C-8b}$$

$$w(y) = 0 \tag{C-8c}$$

for the classical two-dimensional flow.

2. Cylindrical Coordinates

The relationship between velocity and the vector potential in cylindrical coordinates is similar to that relationship just developed for cartesion coordinates. The definition of $\bar{W}$ in cylindrical coordinates is

$$\bar{v} = \nabla x \bar{W} = \begin{vmatrix} \frac{1}{r}\bar{e}_x & \frac{1}{r}\bar{e}_r & \bar{e}_\theta \\ \alpha & D & \beta \\ f & g & rh \end{vmatrix} e^X \tag{C-9}$$

which has the components

$$u(r) = \frac{1}{r}D(rh) - \frac{\beta}{r}g \tag{C-10a}$$

$$v(r) = \frac{\beta}{r}f - \alpha h \tag{C-10b}$$

$$w(r) = \alpha g - Df \tag{C-10c}$$

As before, the components of velocity are related by the continuity equation

$$\nabla \cdot \bar{u} = \alpha u + \frac{1}{r}D(rv) + \frac{\beta}{r}w = 0 \, . \tag{C-11}$$

It is easily shown that the velocity components in cylindrical coordinates also can be represented by two components of the velocity vector potential, without loss of generality other than that imposed by the condition of nondivergence. In the case of β=0 and h(r) = 0, the equations become

$$u(r) = 0 \tag{C-12a}$$

$$v(r) = 0 \tag{C-12b}$$

$$w(r) = \alpha g - Df \tag{C-12c}$$

which cannot be allowed. Boundary conditions in cylindrical coordinates impose the following restriction on β.

$$\beta = in, \qquad n = 0,1,2,\ldots \tag{C-13}$$

The case of n=0 is considered separately from n≠0. For n=0 it is necessary to retain h(r) and set either f(r) or g(r) equal to zero. For the other cases, n≠0, h(r) may be the component of the vector potential which is set to zero. Note that n=0 corresponds to axially symmetric perturbations. For f(r) = 0

$$u(r) = \frac{1}{r}D(rh) - \frac{\beta}{r}g \tag{C-14a}$$

$$v(r) = -\alpha h \tag{C-14b}$$

$$w(r) = \alpha g \, . \tag{C-14c}$$

For g(r) = 0

$$u(r) = \frac{1}{r}D(rh) \tag{C-15a}$$

$$v(r) = \frac{\beta}{r} f - \alpha h \tag{C-15b}$$

$$w(r) = -Df \; . \tag{C-15c}$$

And for $h(r) = 0$

$$u(r) = -\frac{\beta}{r} g \tag{C-16a}$$

$$v(r) = \frac{\beta}{r} f \tag{C-16b}$$

$$w(r) = \alpha g - Df \; . \tag{C-16c}$$

APPENDIX D

DERIVATION OF COEFFICIENTS FOR PLANE POISEUILLE FLOW

The basic equations are

$$-\frac{1}{Re}\nabla\times\nabla\times\bar{\omega} + \nabla\times\bar{V}\times\bar{\omega} - \nabla\times\bar{\Omega}\times\bar{v} - \partial\bar{\omega}/\partial t = 0 \qquad (A-13)$$

$$\bar{\omega} = \nabla\times\bar{v} \qquad (A-8)$$

$$\bar{V} = \bar{i}\frac{3}{2}(1-y^2) = \bar{i}U(y) \qquad (2-3)$$

$$\bar{\Omega} = \bar{k}3y \qquad (2-4)$$

$$\bar{v} = \nabla\times\bar{W} \qquad (2-7)$$

The vector functions $\bar{W}$, $\bar{v}$ and $\bar{\omega}$ are assumed to have solutions of the form of equations (2-9).

$$\bar{W}(x,y,z,t) = [\bar{i}f(y) + \bar{j}g(y) + \bar{k}h(y)]e^{X} \qquad (2-9a)$$

$$\bar{v}(x,y,z,t) = [\bar{i}u(y) + \bar{j}v(y) + \bar{k}w(y)]e^{X} \qquad (2-9b)$$

$$\bar{\omega}(x,y,z,t) = [\bar{i}\xi(y) + \bar{j}\eta(y) + \bar{k}\zeta(y)]e^{X} \qquad (2-9c)$$

where

$$X \equiv \alpha x + \beta z + \gamma t \quad . \qquad (2-10)$$

Equation (A-13) can be written in terms of only $\bar{W}$ by

substituting for $\bar{\omega}$ using equation (A-8) and then substituting for $\bar{v}$ using equation (2-7). The result is

$$-\frac{1}{Re}\nabla x\nabla x\nabla x\nabla x\bar{W} + \nabla x\bar{V}x\nabla x\nabla x\bar{W} - \nabla x\bar{\Omega}x\nabla x\bar{W}$$

$$- \nabla x\nabla x\frac{\partial \bar{W}}{\partial t} = 0 \qquad \text{(D-1)}$$

From the assumed form of the solution for the velocity vector potential $\bar{W}$, given in equations (2-9a) and (2-10), it can be seen that the partial derivatives $\partial/\partial x$, $\partial/\partial z$, and $\partial/\partial t$ become multiplication by α, β and γ, respectively. The partial derivatives $\partial/\partial y$, $\partial^2/\partial y$, etc. will be represented by the operator symbols D, D^2, etc.

The easiest method for determining the form of the coefficients of equation (D-1) is to represent the operators as matrices multiplying the vector $\bar{W}$ and its derivatives. Brackets are used to indicate a 3 by 3 matrix. Using braces to enclose the components of a vector, the following definitions are assumed;

$$\bar{W} \equiv \begin{Bmatrix} f(y) \\ g(y) \\ h(y) \end{Bmatrix} e^{X} \qquad \text{(D-2)}$$

$$D\bar{W} \equiv \begin{Bmatrix} Df(y) \\ Dg(y) \\ Dh(y) \end{Bmatrix} e^{X} \qquad \text{(D-3)}$$

Higher order derivatives are defined similarly.

It is necessary to determine the matrix operator [CP] which can be used to replace the cross product operator, that is, a matrix [CP] which satisfies the relation

$$[CP]\bar{W} = \bar{t}x[s]\bar{W} \qquad \text{(D-4)}$$

where $\bar{t}$ is an arbitrary vector and $[s]$ is an arbitrary matrix. If the components of $\bar{t}$ are t_1, t_2 and t_3 and the components of $[s]$ are s_{ij}, then

$$[CP]\bar{W} = \begin{Bmatrix} t_1 \\ t_2 \\ t_3 \end{Bmatrix} x \begin{bmatrix} s_{11} & s_{12} & s_{13} \\ s_{21} & s_{22} & s_{23} \\ s_{31} & s_{32} & s_{33} \end{bmatrix} \begin{Bmatrix} f \\ g \\ h \end{Bmatrix} e^X \tag{D-5}$$

The matrix $[CP]$ may be obtained by multiplying vector $\bar{W}$ by matrix $[s]$, then finding the cross product of the resulting vector with vector $\bar{t}$ and, finally, collecting terms in f, g, and h to separate out the vector $\bar{W}$. The result is

$$[CP] = \begin{bmatrix} (t_2 s_{31} - t_3 s_{21}) & (t_2 s_{32} - t_3 s_{22}) & (t_2 s_{33} - t_3 s_{23}) \\ (t_3 s_{11} - t_1 s_{31}) & (t_3 s_{12} - t_1 s_{32}) & (t_3 s_{13} - t_1 s_{33}) \\ (t_1 s_{21} - t_2 s_{11}) & (t_1 s_{22} - t_2 s_{12}) & (t_1 s_{23} - t_2 s_{13}) \end{bmatrix} \tag{D-6}$$

A matrix operator is also needed to replace the curl operator, ∇x.

$$\nabla x \bar{W} = \nabla x \begin{Bmatrix} f\,(y) \\ g\,(y) \\ h\,(y) \end{Bmatrix} e^X = \begin{vmatrix} \bar{i} & \bar{j} & \bar{k} \\ \frac{\partial}{\partial x} & \frac{\partial}{\partial y} & \frac{\partial}{\partial z} \\ fe^X & ge^X & he^X \end{vmatrix}$$

$$= \bar{i}\,(Dh - \beta g)\,e^X + \bar{j}\,(\beta f - \alpha h)\,e^X + \bar{k}\,(\alpha g - Df)\,e^X \tag{D-7}$$

This can be written

$$\nabla x \bar{W} = [A]\bar{W} + [B]D\bar{W} \tag{D-8}$$

where [A] and [B] can be determined by collecting the proper terms.

$$[A] = \begin{bmatrix} 0 & -\beta & 0 \\ \beta & 0 & -\alpha \\ 0 & \alpha & 0 \end{bmatrix} \tag{D-9}$$

$$[B] = \begin{bmatrix} 0 & 0 & 1 \\ 0 & 0 & 0 \\ -1 & 0 & 0 \end{bmatrix} \tag{D-10}$$

This operation defined by equation (D-8) holds for the curl of any vector with the form $\big(u(y), v(y), w(y)\big)e^{x}$, that is, for any curl operation performed in equation (D-1).

To evaluate the third term in equation (D-1) take the cross product of $\bar{\Omega}$ and $\nabla x \bar{W}$, that is,

$$\begin{Bmatrix} 0 \\ 0 \\ 3y \end{Bmatrix} x \left([A]\bar{W} + [B]D\bar{W}\right) .$$

This can be evaluated using equations (D-6) and (D-5). The result is

$$\bar{\Omega} x \nabla x \bar{W} = [C_1]\bar{W} + [C_2]D\bar{W} \tag{D-11}$$

where

$$[C_1] = \begin{bmatrix} -3\beta y & 0 & 3\alpha y \\ 0 & -3\beta y & 0 \\ 0 & 0 & 0 \end{bmatrix} \tag{D-12}$$

$$[C_2] = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 3y \\ 0 & 0 & 0 \end{bmatrix} \tag{D-13}$$

Next operate with $-\nabla\times$.

$$\begin{aligned} -\nabla\times([C_1]\bar{W}+[C_2])D\bar{W} &= -[A]([C_1]\bar{W}+[C_2]D\bar{W}) - [B]D([C_1]\bar{W}+[C_2]D\bar{W}) \\ &= -[A][C_1]\bar{W} - [A][C_2]D\bar{W} - [B][C_1]D\bar{W} \\ &\quad - [B]D[C_1]\bar{W} - [B][C_2]D^2\bar{W} - [B]D[C_2]D\bar{W} \\ &= (-[A][C_1] - [B]D[C_1])\bar{W} \\ &\quad + (-[A][C_2] - [B][C_1] - [B]D[C_2])D\bar{W} \\ &\quad - [B][C_2]D^2\bar{W} \end{aligned} \tag{D-14}$$

Note that $D[C_1]$ means $\frac{\partial}{\partial y}[C_1]$. The coefficient of $D^2\bar{W}$ is equal to the zero matrix, so (D-14) may be written

$$-\nabla\times\bar{\Omega}\times\nabla\bar{W} = [E_1]\bar{W} + [E_2]D\bar{W} \,. \tag{D-15}$$

The matrices $[E_1]$ and $[E_2]$ may be found by simple matrix multiplication and addition.

$$[E_1] = \begin{bmatrix} 0 & -3\beta^2 y & 0 \\ 3\beta^2 y & 0 & -3\alpha\beta y \\ -3\beta & 3\alpha\beta y & 3\alpha \end{bmatrix} \tag{D-16}$$

$$[E_2] = \begin{bmatrix} 0 & 0 & 3\beta y \\ 0 & 0 & 0 \\ -3\beta y & 0 & 0 \end{bmatrix} \qquad \text{(D-17)}$$

To calculate the other terms, it is necessary to first find $\nabla x \nabla x \bar{W}$

$$\nabla x(\nabla x \bar{W}) = \nabla x([A]\bar{W} + [B]D\bar{W})$$

$$=[A][A]\bar{W} + [A][B]D\bar{W} + [B][A]D\bar{W}$$

$$+ [B]D[A]\bar{W} + [B][B]D^2\bar{W} + [B]D[B]D\bar{W} \qquad \text{(D-18)}$$

This may be written

$$\nabla x \nabla x \bar{W} = [E_3]\bar{W} + [E_4]D\bar{W} + [E_5]D^2\bar{W} \qquad \text{(D-19)}$$

where the matrices $[E_3]$, $[E_4]$ and $[E_5]$ are determined from the coefficient matrices of equation (D-18).

$$[E_3] = \begin{bmatrix} -\beta^2 & 0 & \alpha\beta \\ 0 & -(\alpha^2+\beta^2) & 0 \\ \alpha\beta & 0 & -\alpha^2 \end{bmatrix} \qquad \text{(D-20)}$$

$$[E_4] = \begin{bmatrix} 0 & \alpha & 0 \\ \alpha & 0 & \beta \\ 0 & \beta & 0 \end{bmatrix} \qquad \text{(D-21)}$$

$$[E_5] = \begin{bmatrix} -1 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & -1 \end{bmatrix} \qquad \text{(D-22)}$$

The term $-\nabla x \nabla x \frac{\partial \bar{W}}{\partial t}$ can now be evaluted.

$$-\nabla\times\nabla\times\frac{\partial\bar{W}}{\partial t} = -\gamma\nabla\times\nabla\times\bar{W}$$

$$= -\gamma([E_3]\bar{W} + [E_4]D\bar{W} + [E_5]D^2\bar{W}) \qquad \text{(D-23)}$$

To evaluate the term $\nabla\times\bar{V}\times\nabla\times\nabla\times\bar{W}$, first find the matrix operator that is equivalent to the cross product of $\bar{V}$ and $\nabla\times\nabla\times\bar{W}$. Using equations (D-19) and (D-6)

$$\bar{V}\times(\nabla\times\nabla\times\bar{W}) = \begin{Bmatrix} U(y) \\ 0 \\ 0 \end{Bmatrix} \times([E_3]\bar{W}+[E_4]D\bar{W}+[E_5]D^2\bar{W})$$

$$= [C_3]\bar{W} + [C_4]D\bar{W} + [C_5]D^2\bar{W} \qquad \text{(D-24)}$$

where

$$[C_3] = \begin{bmatrix} 0 & 0 & 0 \\ -\alpha\beta U & 0 & \alpha^2 U \\ 0 & -(\alpha^2+\beta^2)U & 0 \end{bmatrix} \qquad \text{(D-25)}$$

$$[C_4] = \begin{bmatrix} 0 & 0 & 0 \\ 0 & -\beta U & 0 \\ \alpha U & 0 & \beta U \end{bmatrix} \qquad \text{(D-26)}$$

$$[C_5] = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & U \\ 0 & 0 & 0 \end{bmatrix} \qquad \text{(D-27)}$$

Now operate with $\nabla\times$.

$$\nabla\times\bar{V}\times\nabla\times\nabla\times\bar{W} = \nabla\times([C_3]\bar{W}+[C_4]D\bar{W}+[C_5]D^2\bar{W})$$

$$= ([A]+[B]D)([C_3]\bar{W}+[C_4]D\bar{W}+[C_5]D^2\bar{W})$$

$$= [E_6]\bar{W} + [E_7]D\bar{W} + [E_8]D^2\bar{W} + [E_9]D^3\bar{W} \qquad \text{(D-28)}$$

where

$$[E_6] = \begin{bmatrix} \alpha\beta^2 U & 3(\alpha^2+\beta^2)y & -\alpha^2\beta U \\ 0 & \alpha(\alpha^2+\beta^2)U & 0 \\ -\alpha^2\beta U & 0 & \alpha^3 U \end{bmatrix} \qquad \text{(D-29)}$$

$$[E_7] = \begin{bmatrix} -3\alpha y & -\alpha^2 U & -3\beta y \\ -\alpha^2 U & 0 & -\alpha\beta U \\ 0 & -\alpha\beta U & 0 \end{bmatrix} \qquad \text{(D-30)}$$

$$[E_8] = \begin{bmatrix} \alpha U & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & \alpha U \end{bmatrix} \qquad \text{(D-31)}$$

$[E_9] = [0]$, the zero matrix. (D-32)

To evaluate the term $-\frac{1}{Re}\nabla\times\nabla\times\nabla\times\nabla\times\bar{W}$, first apply the curl operator to $\nabla\times\nabla\times\bar{W}$ which is given by equations (D-19) through (D-22).

$$\nabla\times(\nabla\times\nabla\times\bar{W}) = ([A]+[B]D)([E_3]\bar{W}+[E_4]D\bar{W}+[E_5]D^2\bar{W})$$

$$= [C_6]\bar{W} + [C_7]D\bar{W} + [C_8]D^2\bar{W} + [C_9]D^3\bar{W} \qquad \text{(D-33)}$$

where

$$[C_6] = \begin{bmatrix} 0 & \beta(\alpha^2+\beta^2) & 0 \\ -\beta(\alpha^2+\beta^2) & 0 & \alpha(\alpha^2+\beta^2) \\ 0 & -\alpha(\alpha^2+\beta^2) & 0 \end{bmatrix} \qquad \text{(D-34)}$$

$$[C_7] = \begin{bmatrix} 0 & 0 & -(\alpha^2+\beta^2) \\ 0 & 0 & 0 \\ (\alpha^2+\beta^2) & 0 & 0 \end{bmatrix} \qquad \text{(D-35)}$$

$$[C_8] = \begin{bmatrix} 0 & \beta & 0 \\ -\beta & 0 & \alpha \\ 0 & -\alpha & 0 \end{bmatrix} \qquad \text{(D-36)}$$

$$[C_9] = \begin{bmatrix} 0 & 0 & -1 \\ 0 & 0 & 0 \\ 1 & 0 & 0 \end{bmatrix} \qquad \text{(D-37)}$$

Finally, operate once more with the curl.

$$\nabla x(\nabla x \nabla x \nabla x \bar{W}) = ([A]+[B]D)([C_6]\bar{W}+[C_7]D\bar{W}+[C_8]D^2\bar{W}+[C_9]D^3\bar{W})$$

$$= [E_{10}]\bar{W} + [E_{11}]D\bar{W} + [E_{12}]D^2\bar{W} + [E_{13}]D^3\bar{W}$$

$$+ [E_{14}]D^4\bar{W} \qquad \text{(D-38)}$$

where

$$[E_{10}] = \begin{bmatrix} \beta^2(\alpha^2+\beta^2) & 0 & -\alpha\beta(\alpha^2+\beta^2) \\ 0 & (\alpha^2+\beta^2)^2 & 0 \\ -\alpha\beta(\alpha^2+\beta^2) & 0 & \alpha^2(\alpha^2+\beta^2) \end{bmatrix} \qquad \text{(D-39)}$$

$$[E_{11}] = \begin{bmatrix} 0 & -\alpha(\alpha^2+\beta^2) & 0 \\ -\alpha(\alpha^2+\beta^2) & 0 & -\beta(\alpha^2+\beta^2) \\ 0 & -\beta(\alpha^2+\beta^2) & 0 \end{bmatrix} \qquad \text{(D-40)}$$

$$[E_{12}] = \begin{bmatrix} \alpha^2+2\beta^2 & 0 & -\alpha\beta \\ 0 & (\alpha^2+\beta^2) & 0 \\ -\alpha\beta & 0 & 2\alpha^2+\beta^2 \end{bmatrix} \qquad \text{(D-41)}$$

$$[E_{13}] = \begin{bmatrix} 0 & -\alpha & 0 \\ -\alpha & 0 & -\beta \\ 0 & -\beta & 0 \end{bmatrix} \qquad \text{(D-42)}$$

$$[E_{14}] = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix} \qquad \text{(D-43)}$$

The final equation becomes

$$\begin{aligned} -\frac{1}{Re}[E_{14}]D^4\bar{W} - \frac{1}{Re}[E_{13}]D^3\bar{W} + \left(-\frac{1}{Re}[E_{12}]+[E_8]\right)D^2\bar{W} \\ + \left(-\frac{1}{Re}[E_{11}]+[E_7]+[E_2]\right)D\bar{W} \\ + \left(-\frac{1}{Re}[E_{10}]+[E_6]+[E_1]\right)\bar{W} \\ - \gamma\left([E_5]D^2\bar{W}+[E_4]D\bar{W}+[E_3]\bar{W}\right) = 0 \qquad \text{(D-44)} \end{aligned}$$

which can be written

$$\begin{aligned} [M_4]D^4\bar{W} + [M_3]D^3\bar{W} + [M_2]D^2\bar{W} + [M_1]D\bar{W} + [M_0]\bar{W} \\ + \gamma\left([N_2]D^2\bar{W} + [N_1]D\bar{W} + [N_0]\bar{W}\right) = 0 \qquad \text{(D-45)} \end{aligned}$$

where

$$[M_4] = \begin{bmatrix} -\frac{1}{Re} & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & -\frac{1}{Re} \end{bmatrix} \tag{D-46}$$

$$[M_3] = \begin{bmatrix} 0 & \frac{\alpha}{Re} & 0 \\ \frac{\alpha}{Re} & 0 & \frac{\beta}{Re} \\ 0 & \frac{\beta}{Re} & 0 \end{bmatrix} \tag{D-47}$$

$$[M_2] = \begin{bmatrix} -\frac{\beta^2}{Re}+T & 0 & \frac{\alpha\beta}{Re} \\ 0 & -\frac{1}{Re}(\alpha^2+\beta^2) & 0 \\ \frac{\alpha\beta}{Re} & 0 & -\frac{\alpha^2}{Re}+T \end{bmatrix} \tag{D-48}$$

$$[M_1] = \begin{bmatrix} -3\alpha y & -\alpha T & 0 \\ -\alpha T & 0 & -\beta T \\ -3\beta y & -\beta T & 0 \end{bmatrix} \tag{D-49}$$

$$[M_0] = \begin{bmatrix} \beta^2 T & 3\alpha^2 y & -\alpha\beta T \\ 3\beta^2 y & (\alpha^2+\beta^2)T & -3\alpha\beta y \\ -\alpha\beta T-3\beta & 3\alpha\beta y & 3\alpha+\alpha^2 T \end{bmatrix} \tag{D-50}$$

$$[N_2] = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{D-51}$$

$$[N_1] = \begin{bmatrix} 0 & -\alpha & 0 \\ -\alpha & 0 & -\beta \\ 0 & -\beta & 0 \end{bmatrix} \tag{D-52}$$

$$[N_0] = \begin{bmatrix} \beta^2 & 0 & -\alpha\beta \\ 0 & (\alpha^2+\beta^2) & 0 \\ -\alpha\beta & 0 & \alpha^2 \end{bmatrix} \qquad \text{(D-53)}$$

and T is defined by

$$T \equiv \alpha U - \frac{1}{\overline{Re}}(\alpha^2+\beta^2) = \frac{3}{2}\alpha(1-y^2) - \frac{1}{\overline{Re}}(\alpha^2+\beta^2) \ . \qquad \text{(D-54)}$$

Finally, equation (D-45) can be divided through by e^X to yield the results below.

$$[M_4]\begin{Bmatrix} D^4f \\ D^4g \\ D^4h \end{Bmatrix} + [M_3]\begin{Bmatrix} D^3f \\ D^3g \\ D^3h \end{Bmatrix} + \Big([M_2]+\gamma[N_2]\Big)\begin{Bmatrix} D^2g \\ D^2g \\ D^2h \end{Bmatrix}$$

$$+ \Big([M_1]+\gamma[N_1]\Big)\begin{Bmatrix} Df \\ Dg \\ Dh \end{Bmatrix} + \Big([M_0]+\gamma[N_0]\Big)\begin{Bmatrix} f \\ g \\ h \end{Bmatrix} = 0 \qquad \text{(D-55)}$$

APPENDIX E

DERIVATION OF COEFFICIENTS FOR PIPE FLOW

The basic equations are

$$-\frac{1}{Re}\nabla\times\nabla\times\nabla\times\nabla\times\bar{W} + \nabla\times\bar{V}\times\nabla\times\nabla\times\bar{W} - \nabla\times\bar{\Omega}\times\nabla\times\bar{W} - \nabla\times\nabla\times\frac{\partial\bar{W}}{\partial t} = 0 \qquad \text{(D-1)}$$

$$\bar{V} = \bar{e}_x 2(1-r^2) = \bar{e}_x U(r) \qquad \text{(2-5)}$$

$$\bar{\Omega} = \bar{e}_\theta 4r \qquad \text{(2-6)}$$

and the velocity vector potential $\bar{W}$ has a form similar to that for the plane flow case.

$$\bar{W}(x,r,\theta,t) = [\bar{e}_x f(r) + \bar{e}_r g(r) + \bar{e}_\theta h(r)]e^X \qquad \text{(E-1)}$$

where

$$X \equiv \alpha x + \beta\theta + \gamma t \,. \qquad \text{(E-2)}$$

As in the plane flow case, the partial derivatives $\partial/\partial x$, $\partial/\partial\theta$ and $\partial/\partial t$ become multiplication by α, β and γ, respectively. The partial derivatives $\partial/\partial r$, $\partial^2/\partial r^2$, etc. will be represented by the operator symbols D, D^2, etc.

The method of solution for cylindrical coordinates proceeds exactly as that for the cartesian coordinate case presented in Appendix D. Definitions similar to (D-2) and (D-3) are assumed.

$$\bar{W} \equiv \begin{Bmatrix} f(r) \\ g(r) \\ h(r) \end{Bmatrix} e^X \qquad \text{(E-3)}$$

$$D\bar{W} = \begin{Bmatrix} Df(r) \\ Dg(r) \\ Dh(r) \end{Bmatrix} e^{X} \tag{E-4}$$

Higher order derivatives are similarly defined. The matrix [CP] defined by equation (D-6) is valid for cylindrical coordinates and is used to evaluate cross products. The curl operation in cylindrical coordinates is found below.

$$\nabla \times \bar{W} = \nabla \times \begin{Bmatrix} f(r) \\ g(r) \\ h(r) \end{Bmatrix} e^{X} = \begin{vmatrix} \frac{1}{r}\bar{e}_x & \frac{1}{r}\bar{e}_r & \bar{e}_\theta \\ \frac{\partial}{\partial x} & \frac{\partial}{\partial r} & \frac{\partial}{\partial \theta} \\ fe^{X} & ge^{X} & rhe^{X} \end{vmatrix}$$

$$= \frac{1}{r}\bar{e}_x\left(h+rDh-\beta g\right)e^{X} + \frac{1}{r}\bar{e}_r\left(\beta f-r\alpha h\right)e^{X} + \bar{e}_\theta\left(\alpha g-Df\right)e^{X} \tag{E-5}$$

This can be written

$$\nabla \times \bar{W} = [A]\bar{W} + [B]D\bar{W} \tag{E-6}$$

where

$$[A] = \begin{bmatrix} 0 & -\frac{\beta}{r} & \frac{1}{r} \\ \frac{\beta}{r} & 0 & -\alpha \\ 0 & \alpha & 0 \end{bmatrix} \tag{E-7}$$

$$[B] = \begin{bmatrix} 0 & 0 & 1 \\ 0 & 0 & 0 \\ -1 & 0 & 0 \end{bmatrix} \tag{E-8}$$

Following the method of Appendix D, the first step is to use equation (D-6) to evaluate $\bar{\Omega} \times \nabla \times \bar{W}$.

$$\bar{\Omega} \times \nabla \times \bar{W} = \begin{Bmatrix} 0 \\ 0 \\ 4r \end{Bmatrix} \times \left([A]+[B]D\bar{W}\right) = [C_1]\bar{W} + [C_2]D\bar{W} \tag{E-9}$$

where

$$[C_1] = \begin{bmatrix} -4\beta & 0 & 4\alpha r \\ 0 & -4\beta & 4 \\ 0 & 0 & 0 \end{bmatrix} \tag{E-10}$$

$$[C_2] = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 4r \\ 0 & 0 & 0 \end{bmatrix} \tag{E-11}$$

Next, operate with $-\nabla\times$

$$-\nabla\times\bar{\Omega}\times\nabla\times\bar{W} = -([A]+[B]D)([C_1]\bar{W}+[C_2]D\bar{W})$$

$$=[E_1]\bar{W} + [E_2]D\bar{W} + [0]D^2\bar{W} \tag{E-12}$$

where

$$[E_1] = \begin{bmatrix} 0 & -\frac{4\beta^2}{r} & \frac{4\beta}{r} \\ \frac{4\beta^2}{r} & 0 & -4\alpha\beta \\ 0 & 4\alpha\beta & 0 \end{bmatrix} \tag{E-13}$$

$$[E_2] = \begin{bmatrix} 0 & 0 & 4\beta \\ 0 & 0 & 0 \\ -4\beta & 0 & 0 \end{bmatrix} \tag{E-14}$$

Next, find $\nabla\times\nabla\times\bar{W}$.

$$\nabla\times(\nabla\times\bar{W}) = ([A]+[B]D)([A]\bar{W}+[B]D\bar{W})$$

$$= [E_3]\bar{W} + [E_4]D\bar{W} + [E_5]D^2\bar{W} \tag{E-15}$$

where

$$[E_3] = \begin{bmatrix} -\frac{\beta^2}{r^2} & \frac{\alpha}{r} & \frac{\alpha\beta}{r} \\ 0 & -(\alpha^2+\frac{\beta^2}{r^2}) & \frac{\beta}{r^2} \\ \frac{\alpha\beta}{r} & -\frac{\beta}{r^2} & -\alpha^2+\frac{1}{r^2} \end{bmatrix} \qquad \text{(E-16)}$$

$$[E_4] = \begin{bmatrix} -\frac{1}{r} & \alpha & 0 \\ \alpha & 0 & -\frac{\beta}{r} \\ 0 & \frac{\beta}{r} & -\frac{1}{r} \end{bmatrix} \qquad \text{(E-17)}$$

$$[E_5] = \begin{bmatrix} -1 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & -1 \end{bmatrix} \qquad \text{(E-18)}$$

The term $-\nabla x \nabla x \frac{\partial \bar{W}}{\partial t}$ can now be evaluated.

$$-\nabla x \nabla x \frac{\partial \bar{W}}{\partial t} = -\gamma \nabla x \nabla x \bar{W}$$

$$= -\gamma\Big([E_3]\bar{W} + [E_4]D\bar{W} + [E_5]D^2\bar{W}\Big) \qquad \text{(E-19)}$$

Next find $\bar{V}x\nabla x \nabla x \bar{W}$ using equations (E-15) and (D-6).

$$\bar{V}x(\nabla x \nabla x \bar{W}) = \begin{Bmatrix} U(r) \\ 0 \\ 0 \end{Bmatrix} x([E_3]\bar{W}+[E_4]D\bar{W}+[E_5]D^2\bar{W})$$

$$= [C_3]\bar{W} + [C_4]D\bar{W} + [C_5]D^2\bar{W} \qquad \text{(E-20)}$$

where

$$[C_3] = \begin{bmatrix} 0 & 0 & 0 \\ -\frac{\alpha\beta}{r}U & \frac{\beta}{r^2}U & (\alpha^2-\frac{1}{r^2})U \\ 0 & -(\alpha^2+\frac{\beta^2}{r^2})U & \frac{\beta}{r^2}U \end{bmatrix} \qquad \text{(E-21)}$$

$$[C_4] = \begin{bmatrix} 0 & 0 & 0 \\ 0 & -\frac{\beta}{r}U & \frac{1}{r}U \\ \alpha U & 0 & \frac{\beta}{r}U \end{bmatrix} \tag{E-22}$$

$$[C_5] = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & U \\ 0 & 0 & 0 \end{bmatrix} \tag{E-23}$$

Now, operating with ∇x, the result is

$$\nabla x \bar{V} x \nabla x \nabla x \bar{W} = \nabla x ([C_3]\bar{W} + [C_4]D\bar{W} + [C_5]D^2\bar{W})$$

$$= [E_6]\bar{W} + [E_7]D\bar{W} + [E_8]D^2\bar{W} + [E_9]D^3\bar{W} \tag{E-24}$$

where

$$[E_6] = \begin{bmatrix} \frac{\alpha\beta^2}{r^2}U & 4r(\alpha^2+\frac{\beta^2}{r^2}) - \frac{\alpha^2}{r}U & -4\frac{\beta^2}{r} - \beta\frac{\alpha^2}{r}U \\ 0 & \alpha(\alpha^2+\frac{\beta^2}{r^2})U & -\frac{\alpha\beta}{r^2}U \\ -\frac{\alpha^2\beta U}{r} & \frac{\alpha\beta}{r^2}U & (\alpha^3-\frac{\alpha}{r})U \end{bmatrix} \tag{E-25}$$

$$[E_7] = \begin{bmatrix} \frac{\alpha}{r}U - 4r\alpha & -\alpha^2 U & -4\beta \\ -\alpha^2 U & 0 & -\frac{\alpha\beta}{r}U \\ 0 & -\frac{\alpha\beta}{r}U & \frac{\alpha}{r}U \end{bmatrix} \tag{E-26}$$

$$[E_8] = \begin{bmatrix} \alpha U & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & \alpha U \end{bmatrix} \tag{E-27}$$

and the matrix $[E_9]$ is identically zero. For the final term, first calculate $\nabla x \nabla x \nabla x \bar{W}$.

$$\nabla\times(\nabla\times\nabla\times\bar{W}) = ([A]+[B]D)([E_3]\bar{W}+[E_4]D\bar{W}+[E_5]D^2\bar{W})$$

$$= [C_6]\bar{W} + [C_7]D\bar{W} + [C_8]D^2\bar{W} + [C_9]D^3\bar{W} \qquad \text{(E-28)}$$

where

$$[C_6] = \begin{bmatrix} 0 & \frac{\beta}{r^3}+\frac{\beta}{r}(\alpha^2+\frac{\beta^2}{r^2}) & -\frac{1}{r^3}-\frac{1}{r}(\alpha^2+\frac{\beta^2}{r^2}) \\ -\frac{\beta}{r}(\alpha^2+\frac{\beta^2}{r^2}) & \frac{2\alpha\beta}{r^2} & -\frac{\alpha}{r^2}+\alpha(\alpha^2+\frac{\beta^2}{r^2}) \\ -2\frac{\beta^2}{r^3} & \frac{\alpha}{r^2}-\alpha(\alpha^2+\frac{\beta^2}{r^2}) & 2\frac{\alpha\beta}{r^2} \end{bmatrix} \qquad \text{(E-29)}$$

$$[C_7] = \begin{bmatrix} 0 & -\frac{\beta}{r^2} & -(\alpha^2+\frac{\beta^2}{r^2})+\frac{1}{r^2} \\ -\frac{\beta}{r^2} & 0 & \frac{\alpha}{r} \\ (\alpha^2+\frac{\beta^2}{r^2})-\frac{1}{r^2} & -\frac{\alpha}{r} & 0 \end{bmatrix} \qquad \text{(E-30)}$$

$$[C_8] = \begin{bmatrix} 0 & \frac{\beta}{r} & -\frac{2}{r} \\ -\frac{\beta}{r} & 0 & \alpha \\ \frac{1}{r} & -\alpha & 0 \end{bmatrix} \qquad \text{(E-31)}$$

$$[C_9] = \begin{bmatrix} 0 & 0 & -1 \\ 0 & 0 & 0 \\ 1 & 0 & 0 \end{bmatrix} \qquad \text{(E-32)}$$

And, finally,

$$\nabla\times(\nabla\times\nabla\times\nabla\times\bar{W}) = ([A]+[B]D)([C_6]\bar{W}+[C_7]D\bar{W}+[C_8]D^2\bar{W}+[C_9]D^3\bar{W})$$

$$= [E_{10}]\bar{W} + [E_{11}]D\bar{W} + [E_{12}]D^2\bar{W} + [E_{13}]D^3\bar{W}$$

$$+ [E_{14}]D^4\bar{W} \qquad \text{(E-33)}$$

Letting

$$t \equiv \frac{(\alpha^2+\beta^2)}{r^2} \qquad \text{(E-34)}$$

the coefficients are

$$[E_{10}] = \begin{bmatrix} \frac{\beta^2}{r^2}t+4\frac{\beta^2}{r^4} & -\frac{\alpha}{r^3}-\frac{\alpha}{r}t & -\frac{\alpha\beta}{r^3}-\frac{\alpha\beta}{r}t \\ 2\frac{\alpha\beta^2}{r^3} & \frac{\beta^2}{r^4}-\frac{\alpha^2}{r^2}+t^2 & -\frac{\beta}{r^4}-2\frac{\alpha^2\beta}{r^2}-\frac{\beta}{r^2}t \\ -\frac{\alpha\beta}{r}t & \frac{3\beta}{r^4}+\frac{3\beta}{r^2}t & -\frac{3}{r^4}-2\frac{\alpha^2}{r^2}-3\frac{\beta^2}{r^4}+\alpha^2 t \end{bmatrix} \qquad \text{(E-35)}$$

$$[E_{11}] = \begin{bmatrix} -3\frac{\beta^2}{r^3}+\frac{1}{r^3}+\frac{t}{r} & \frac{\alpha}{r^2}-\alpha t & \frac{\alpha\beta}{r^2} \\ -\alpha t+\frac{\alpha}{r^2} & -\frac{\beta^2}{r^3}+\frac{\alpha^2}{r} & -\frac{\beta}{r}t+\frac{\beta}{r^3} \\ -\frac{\alpha\beta}{r^2} & -\frac{3\beta}{r^3}-\frac{\beta}{r}t & \frac{3}{r^3}+\frac{\alpha^2}{r}-2\frac{\beta^2}{r^3}+\frac{t}{r} \end{bmatrix} \qquad \text{(E-36)}$$

$$[E_{12}] = \begin{bmatrix} \frac{\beta^2}{r^2}-\frac{1}{r^2}+t & -2\frac{\alpha}{r} & -\frac{\alpha\beta}{r} \\ -\frac{\alpha}{r} & t & -\frac{2\beta}{r^2} \\ -\frac{\alpha\beta}{r} & \frac{2\beta}{r^2} & \alpha^2-\frac{3}{r^2}+t \end{bmatrix} \qquad \text{(E-37)}$$

$$[E_{13}] = \begin{bmatrix} \frac{2}{r} & -\alpha & 0 \\ -\alpha & 0 & -\frac{\beta}{r} \\ 0 & -\frac{\beta}{r} & \frac{2}{r} \end{bmatrix} \qquad \text{(E-38)}$$

$$[E_{14}] = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix} \qquad \text{(E-39)}$$

The final equation, with all the terms grouped together has the same form as equation (D-43) and, dividing by e^X,

can be written in the form of equation (D-55).

$$[M_4]\begin{Bmatrix} D^4 f \\ D^4 g \\ D^4 h \end{Bmatrix} + [M_3]\begin{Bmatrix} D^3 f \\ D^3 g \\ D^3 h \end{Bmatrix} + ([M_2]+\gamma[N_2])\begin{Bmatrix} D^2 g \\ D^2 g \\ D^2 h \end{Bmatrix}$$

$$+ ([M_1]+\gamma[N_1])\begin{Bmatrix} Df \\ Dg \\ Dh \end{Bmatrix} + ([M_0]+\gamma[N_0])\begin{Bmatrix} f \\ g \\ h \end{Bmatrix} = 0 \qquad \text{(D-55)}$$

where

$$[M_4] = \begin{bmatrix} -\frac{1}{Re} & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & -\frac{1}{Re} \end{bmatrix} \qquad \text{(E-40)}$$

$$[M_3] = \begin{bmatrix} -\frac{2}{rRe} & \frac{\alpha}{Re} & 0 \\ \frac{\alpha}{Re} & 0 & \frac{\beta}{rRe} \\ 0 & \frac{\beta}{rRe} & -\frac{2}{rRe} \end{bmatrix} \qquad \text{(E-41)}$$

$$[M_2] = \begin{bmatrix} T+\frac{1}{r^2Re}-\frac{\beta^2}{r^2Re} & \frac{2\alpha}{rRe} & \frac{\alpha\beta}{rRe} \\ \frac{\alpha}{rRe} & -\frac{t}{Re} & \frac{2\beta}{r^2Re} \\ \frac{\alpha\beta}{rRe} & -\frac{2\beta}{r^2Re} & T+\frac{3}{r^2Re}-\frac{\alpha^2}{Re} \end{bmatrix} \qquad \text{(E-42)}$$

$$[M_1] = \begin{bmatrix} \frac{1}{r}T-4r\alpha+\frac{3\beta^2}{r^3Re}-\frac{1}{r^3Re} & -\alpha T-\frac{\alpha}{r^2Re} & -\frac{\alpha\beta}{r^2Re} \\ -\alpha T-\frac{\alpha}{r^2Re} & \frac{\beta^2}{r^3Re}-\frac{\alpha^2}{rRe} & -\frac{\beta}{r}T-\frac{\beta}{r^3Re} \\ \frac{\alpha\beta}{r^2Re}-4\beta & -\frac{\beta}{r}T+\frac{3\beta}{r^3Re} & \frac{1}{r}T+\frac{2\beta^2}{r^3Re}-\frac{3}{r^3Re}-\frac{\alpha^2}{rRe} \end{bmatrix} \qquad \text{(E-43)}$$

$$[M_0] =$$

$$\begin{bmatrix} \frac{\beta^2}{r^2}T - \frac{4\beta^2}{r^4 Re} & -\frac{\alpha}{r}T + 4r\alpha^2 + \frac{\alpha}{r^3 Re} & -\frac{\alpha\beta}{r}T + \frac{\alpha\beta}{r^3 Re} \\ \frac{4\beta^2}{r} - \frac{2\alpha\beta^2}{r^3 Re} & tT + \frac{\alpha^2}{r^2 Re} - \frac{\beta^2}{r^4 Re} & -\frac{\beta}{r^2}T - 4\alpha\beta + \frac{\beta}{r^4 Re} + \frac{2\alpha^2\beta}{r^2 Re} \\ -\frac{\alpha}{r}\beta T & \frac{\beta}{r^2}T + 4\alpha\beta - \frac{3\beta}{r^4 Re} - \frac{2\beta t}{r^2 Re} & (\alpha^2 - \frac{1}{r^2})T + \frac{3}{r^4 Re} + \frac{\alpha^2}{r^2 Re} + \frac{2\beta^2}{r^4 Re} \end{bmatrix} \tag{E-44}$$

$$[N_2] = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{E-45}$$

$$[N_1] = \begin{bmatrix} \frac{1}{r} & -\alpha & 0 \\ -\alpha & 0 & -\frac{\beta}{r} \\ 0 & -\frac{\beta}{r} & \frac{1}{r} \end{bmatrix} \tag{E-46}$$

$$[N_0] = \begin{bmatrix} \frac{\beta^2}{r^2} & -\frac{\alpha}{r} & -\frac{\alpha\beta}{r} \\ 0 & t & -\frac{\beta}{r^2} \\ -\frac{\alpha\beta}{r} & \frac{\beta}{r^2} & \alpha^2 - \frac{1}{r^2} \end{bmatrix} \tag{E-47}$$

and

$$T \equiv \alpha U - \frac{t}{Re} = 2\alpha(1-r^2) - \frac{1}{Re}(\alpha^2 + \frac{\beta^2}{r^2}) \tag{E-48}$$

and t is defined by equation (E-34).

APPENDIX F

BOUNDARY CONDITIONS AT THE WALL

1. Boundary Conditions for Plane Flow at y=±1

The boundary condition is that the velocity $\bar{v}$ is zero at the walls. The boundary conditions for $\bar{W}$ must be determined. By the definition of $\bar{W}$,

$$\bar{v} = \nabla \times \bar{W} \ . \tag{2-7}$$

The form of $\bar{v}$ and $\bar{W}$ is given by equations (2-9a), (2-9b) and (2-10). Equation (2-7) can be written

$$\begin{Bmatrix} u(y)\,e^{X} \\ v(y)\,e^{X} \\ w(y)\,e^{X} \end{Bmatrix} = \nabla \times \begin{Bmatrix} f(y)\,e^{X} \\ g(y)\,e^{X} \\ h(y)\,e^{X} \end{Bmatrix}$$

$$= \begin{vmatrix} \bar{i} & \bar{j} & \bar{k} \\ \partial/\partial x & \partial/\partial y & \partial/\partial z \\ f(y)\,e^{X} & g(y)\,e^{X} & h(y)\,e^{X} \end{vmatrix}$$

$$= \begin{vmatrix} \bar{i} & \bar{j} & \bar{k} \\ \alpha & D & \beta \\ f(y) & g(y) & h(y) \end{vmatrix} e^{X}$$

$$= \bar{i}\,(Dh-\beta g)\,e^{X} + \bar{j}\,(\beta f-\alpha h)\,e^{X} + \bar{k}\,(\alpha g-Df)\,e^{X} \tag{F-1}$$

Equation (F-1) must be satisfied by each component

separately. Dividing by e^{X}, the results are

$$u(y) = Dh(y) - \beta g(y) \tag{F-2a}$$

$$v(y) = \beta f(y) - \alpha h(y) \tag{F-2b}$$

$$w(y) = \alpha g(y) - Df(y) \tag{F-2c}$$

As described in Appendix C, one component of the velocity vector potential $\bar{W}$ may be arbitrarily set equal to zero.

a. Case 1... $f(y) = 0$

At $y = \pm 1$, equations (F-2) become

$$u(\pm 1) = 0 = Dh(\pm 1) - \beta g(\pm 1) \tag{F-3a}$$

$$v(\pm 1) = 0 = -\alpha h(\pm 1) \tag{F-3b}$$

$$w(\pm 1) = 0 = \alpha g(\pm 1) \ . \tag{F-3c}$$

These equations imply

$$g(\pm 1) = 0 \tag{F-4a}$$

$$h(\pm 1) = 0 \tag{F-4b}$$

$$Dh(\pm 1) = 0 \ . \tag{F-4c}$$

b. Case 2... $g(y) = 0$

At $y = \pm 1$, equations (F-2) become

$$u(\pm 1) = 0 = Dh(\pm 1) \tag{F-5a}$$

$$v(\pm 1) = 0 = \beta f(\pm 1) - \alpha h(\pm 1) \tag{F-5b}$$

$$w(\pm 1) = 0 = -Df(\pm 1) \ . \tag{F-5c}$$

These equations imply

$$\beta f(\pm 1) = \alpha h(\pm 1) \tag{F-6a}$$

$$Df(\pm 1) = 0 \tag{F-6b}$$

$$Dh(\pm 1) = 0 \; . \tag{F-6c}$$

c. Case 3... $h(y) = 0$

At $y = \pm 1$, equations (F-2) become

$$u(\pm 1) = -\beta g(\pm 1) \tag{F-7a}$$

$$v(\pm 1) = \beta f(\pm 1) \tag{F-7b}$$

$$w(\pm 1) = \alpha g(\pm 1) - Df(\pm 1) \; . \tag{F-7c}$$

These equations imply

$$f(\pm 1) = 0 \tag{F-8a}$$

$$Df(\pm 1) = 0 \tag{F-8b}$$

$$g(\pm 1) = 0 \; . \tag{F-8c}$$

2. <u>Boundary Conditions for Pipe Flow at r = 1</u>

The boundary condition is that the velocity $\bar{v}$ is zero at the wall. The boundary conditions for $\bar{W}$ must be determined. By the definition of $\bar{W}$,

$$\bar{v} = \nabla \times \bar{W} \; . \tag{2-7}$$

The form of $\bar{v}$ and $\bar{W}$ is

$$\bar{v}(x,r,\theta,t) = (\bar{e}_x u(r) + \bar{e}_r v(r) + \bar{e}_\theta w(r)) e^X \tag{F-9}$$

$$\bar{W}(x,r,\theta,t) = (\bar{e}_x f(r) + \bar{e}_r g(r) + \bar{e}_\theta h(r)) e^X \tag{F-10}$$

where

$$X \equiv \alpha x + \beta\theta + \gamma t \, . \tag{F-11}$$

In cylindrical coordinates, equation (2-7) can be written

$$\begin{Bmatrix} u(r)e^X \\ v(r)e^X \\ w(r)e^X \end{Bmatrix} = \nabla \times \begin{Bmatrix} f(r)e^X \\ g(r)e^X \\ h(r)e^X \end{Bmatrix}$$

$$= \begin{vmatrix} \frac{1}{r}\bar{e}_x & \frac{1}{r}\bar{e}_r & \bar{e}_\theta \\ \frac{\partial}{\partial x} & \frac{\partial}{\partial r} & \frac{\partial}{\partial \theta} \\ fe^X & ge^X & rhe^X \end{vmatrix} = \begin{vmatrix} \frac{1}{r}\bar{e}_x & \frac{1}{r}\bar{e}_r & \bar{e}_\theta \\ \alpha & D & \beta \\ f & g & rh \end{vmatrix} e^X$$

$$= \frac{1}{r}\bar{e}_x \left(D(rh) - \beta g\right) e^X + \frac{1}{r}\bar{e}_r \left(\beta f - \alpha r h\right) e^X + \bar{e}_\theta \left(\alpha g - Df\right) \tag{F-12}$$

Equation (F-12) must be satisfied by each component separately. Dividing by e^X, the results are

$$\begin{aligned} u(r) &= \frac{1}{r}D(rh(r)) - \frac{\beta}{r}g(r) \\ &= \frac{1}{r}h(r) + Dh(r) - \frac{\beta}{r}g(r) \end{aligned} \tag{F-13a}$$

$$v(r) = \frac{\beta}{r}f(r) - \alpha h(r) \tag{F-13b}$$

$$w(r) = \alpha g(r) - Df(r) \tag{F-13c}$$

As described in Appendix C, one component of $\bar{W}$ may be arbitrarily set equal to zero.

a. Case 1... $f(r) = 0$

At r=1, equations (F-13) become

$$u(1) = 0 = h(1) + Dh(1) - \beta g(1) \tag{F-14a}$$

$$v(1) = 0 = -\alpha h(1) \tag{F-14b}$$

$$w(1) = 0 = \alpha g(1) \tag{F-14c}$$

These equations imply

$$g(1) = 0 \tag{F-15a}$$

$$h(1) = 0 \tag{F-15b}$$

$$Dh(1) = 0 \tag{F-15c}$$

b. Case 2... $g(r) = 0$

At r=1, equations (F-13) become

$$u(1) = 0 = h(1) + Dh(1) \tag{F-16a}$$

$$v(1) = 0 = \beta f(1) - \alpha h(1) \tag{F-16b}$$

$$w(1) = 0 = -Df(1) \tag{F-16c}$$

These equations imply

$$\beta f(1) = \alpha h(1) \tag{F-17a}$$

$$Df(1) = 0 \tag{F-17b}$$

$$h(1) = -Dh(1) \tag{F-17c}$$

c. Case 3... $h(r) = 0$

At $r=1$, equations (F-13) become

$$u(1) = 0 = -\beta g(1) \tag{F-18a}$$

$$v(1) = 0 = \beta f(1) \tag{F-18b}$$

$$w(1) = 0 = \alpha g(1) - Df(1) \tag{F-18c}$$

These equations imply

$$g(1) = 0 \tag{F-19a}$$

$$f(1) = 0 \tag{F-19b}$$

$$Df(1) = 0 \tag{F-19c}$$

APPENDIX G

BOUNDARY CONDITIONS
FOR PIPE FLOW AT r = 0.
PERIODICITY OF θ.

1. Boundary condition for θ

The condition imposed is that functions be single valued at θ=0 and θ=2π. The assumed form of the solution is

$$\bar{W}(x,r,\theta,t) = [\bar{e}_x f(r) + \bar{e}_r g(r) + \bar{e}_\theta h(r)]e^{X} \qquad \text{(E-1)}$$

where

$$X \equiv \alpha x + \beta\theta + \gamma t \;. \qquad \text{(E-2)}$$

Thus the θ dependence may be separated as the factor

$$e^{\beta\theta} = \cos(-i\beta\theta) + i\sin(-i\beta\theta) \qquad \text{(G-1)}$$

This term must be single valued at θ=0 and θ=2π. Both the real and imaginary parts must match. This results in two equations.

$$\cos(-i\beta 2\pi) = 1 \qquad \text{(G-2a)}$$

$$\sin(-i\beta 2\pi) = 0 \qquad \text{(G-2b)}$$

These conditions are met only for arguments of sine and cosine which are even multiples of π, that is

$$(-i\beta 2\pi) = 2n\pi, \quad n=0,1,2,... \qquad \text{(G-3)}$$

or

$$\beta = ni, \quad n=0,1,2,... \qquad \text{(G-4)}$$

2. <u>Single-Valuedness at r=0</u>

It is required that a vector function with the form of equation (2-9) be single-valued at the origin, r=0. Consider a vector $\bar{V}_n$ which has the form

$$\bar{V}_n = \left(\bar{e}_x u_n(r) + \bar{e}_r v_n(r) + \bar{e}_\theta w_n(r)\right) e^X \tag{G-5}$$

where

$$X = \alpha x + in\theta + \gamma t\ . \tag{G-6}$$

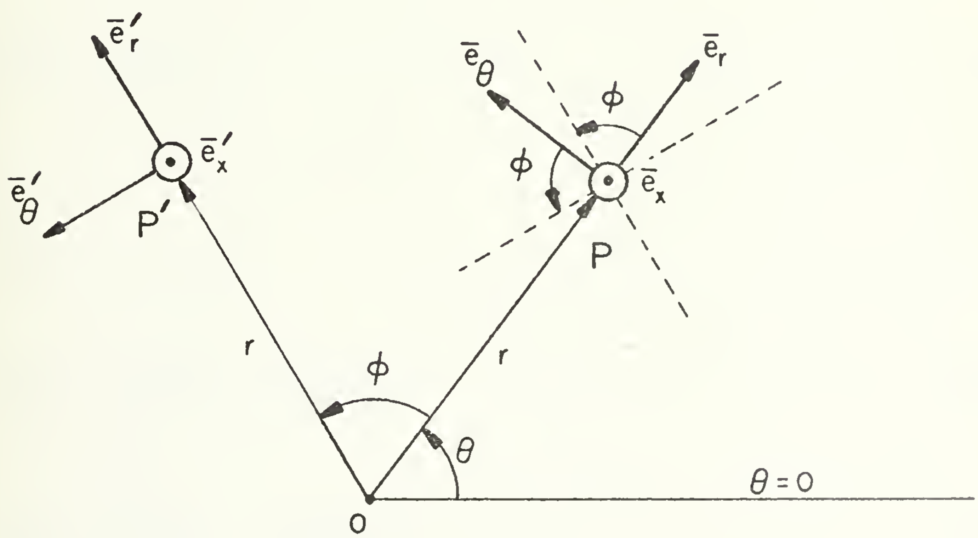

Figure G-1 Vector Function Rotated ϕ

Consider figure G-1 where point P is rotated an angle ϕ about the origin to become point P', both at a distance r from the origin. The unit coordinate vectors are $\bar{e}_x$, $\bar{e}_r$, and $\bar{e}_\theta$ at P and $\bar{e}'_x$, $\bar{e}'_r$, and $\bar{e}'_\theta$ at P'. Vectors $\bar{e}_x$ and $\bar{e}'_x$ extend out of the picture normal to $\bar{e}_r$ and $\bar{e}_\theta$. At point P', $\bar{V}_n$ becomes

$$\bar{V}'_n = \left(\bar{e}'_x u_n(r) + \bar{e}'_r v_n(r) + \bar{e}'_\theta w_n(r)\right) e^{X'} \tag{G-7}$$

where

$$X' = \alpha x + in(\theta+\phi) + \gamma t \,. \tag{G-8}$$

Then

$$e^{X'} = e^{X} e^{in\phi} \tag{G-9}$$

and

$$\bar{e}'_x = \bar{e}_x \tag{G-10a}$$

$$\bar{e}'_r = \bar{e}_r \cos\phi + \bar{e}_\theta \sin\phi \tag{G-10b}$$

$$\bar{e}'_\theta = \bar{e}_\theta \cos\phi - \bar{e}_r \sin\phi \tag{G-10c}$$

Substituting (G-9) and (G-10) into (G-7) and rearranging terms results in equation (G-11).

$$\begin{aligned}\bar{V}'_n = {} & \bar{e}_x u_n(r) + \bar{e}_r\left(v_n(r)\cos\phi - w_n(r)\sin\phi\right) \\ & + \bar{e}_\theta\left(v_n(r)\sin\phi + w_n(r)\cos\phi\right) e^{X} e^{in\phi}\end{aligned} \tag{G-11}$$

It is required that

$$\lim_{r \to 0} \bar{V}_n = \lim_{r \to 0} \bar{V}'_n . \qquad \text{(G-12)}$$

If equations (G-11) and (G-5) are set equal to each other at r=0, then each component of this vector equation must separately satisfy the equality. The result is the following three equations.

$$u_n(0) = u_n(0)e^{in\phi} \qquad \text{(G-13)}$$

$$v(0) = [v_n(0)\cos\phi - w_n(0)\sin\phi]e^{in\phi} \qquad \text{(G-14)}$$

$$w_n(0) = [v_n(0)\sin\phi + w_n(0)\cos\phi]e^{in\phi} \qquad \text{(G-15)}$$

Equation (G-13) can be written

$$u_n(0)[1-e^{in\phi}] = 0 . \qquad \text{(G-16)}$$

For $n=0$, $1-e^{in\phi}$ is identically zero, so $u_0(0)$ is unrestricted. For $n\neq0$, $1-e^{in\phi}$ is, in general, not equal to zero so $u_n(0)=0$ for $n\neq0$. Equations (G-14) and (G-15) are rewritten

$$v_n(0)(1-\cos\phi e^{in\phi}) + w_n(0)(\sin\phi e^{in\phi}) = 0 \qquad \text{(G-17)}$$

$$v_n(0)(-\sin\phi e^{in\phi}) + w_n(0)(1-\cos\phi e^{in\phi}) = 0 \qquad \text{(G-18)}$$

In (G-17) and (G-18), $v_n(0)$ and $w_n(0)$ must equal zero unless the determinant of the coefficients vanishes. Setting the determinant of the coefficients equal to zero yields the following equation.

$$(1-\cos\phi e^{in\phi})^2 + (\sin\phi e^{in\phi})^2 = 0 \quad \text{(G-19)}$$

Multipling the terms out, using Eulers equation and noting that $\sin^2\phi+\cos^2\phi=1$, results in

$$1 - 2\cos\phi(\cos n\phi+i\sin n\phi) + \cos 2n\phi + i\sin 2n\phi = 0 \quad \text{(G-20)}$$

The real and imaginary parts of equation (G-20) must separately be satisfied as follows.

$$1 - 2\cos\phi\cos n\phi + \cos 2n\phi = 0 \quad \text{(G-21)}$$

$$-2\cos\phi\sin n\phi + \sin 2n\phi = 0 \quad \text{(G-22)}$$

Using well known identities for $\cos 2n\phi$ and $\sin 2n\phi$ yields the following expressions.

$$1 + \cos n\phi(\cos n\phi-2\cos\phi) -\sin^2 n\phi = 0 \quad \text{(G-23)}$$

$$\sin n\phi(\cos\phi-\cos n\phi) = 0 \quad \text{(G-24)}$$

For $n=0$, equation (G-24) is satisfied identically, but equation (G-23) reduces to

$$2(1-\cos\phi) = 0 \quad \text{(G-25)}$$

which, in general, is not satisfied for arbitrary ϕ. For $n=1$, equations (G-23) and (G-24) become

$$1 - (\sin^2\phi+\cos^2\phi) = 0 \quad \text{(G-26)}$$

$$\sin\phi(\cos\phi-\cos\phi) = 0 \quad \text{(G-27)}$$

which are satisfied identically for any value of ϕ. For $n>1$, the equations will not be satisfied. In particular, equation (G-24) requires that either $\sin n\phi=0$ or $\cos\phi=\cos n\phi$

which cannot be satisfied for arbitrary ϕ and for $n>1$.

A relationship between $v_1(0)$ and $w_1(0)$ is implied by the previous result. Using equation (G-17) to find the ratio of $v_1(0)$ and $w_1(0)$ results in

$$\frac{v_1(0)}{w_1(0)} = -\frac{\sin\phi e^{i\phi}}{1-\cos\phi e^{i\phi}} = -\frac{\sin\phi}{e^{-i\phi}-\cos\phi}$$

$$= \frac{-\sin\phi}{\cos(-\phi) + i\sin(-\phi) - \cos\phi} = i \tag{G-28}$$

or

$$v_1(0) = iw_1(0) \tag{G-29}$$

Equation (G-18) yields the same result. A summary of the results of applying the restriction that a vector function of the form specified by equations (G-5) and (G-6) be single valued at $r=0$ fcllcws.

$n=0$ $u_0(0)$ unrestricted (G-30a)

$$v_0(0) = 0 \tag{G-30b}$$

$$w_0(0) = 0 \tag{G-30c}$$

$$n=1 \quad u_1(0) = 0 \tag{G-31a}$$

$$v_1(0) = iw_1(0) \tag{G-31b}$$

$$n>1 \quad u_n(0) = 0 \tag{G-32a}$$

$$v_n(0) = 0 \tag{G-32b}$$

$$w_n(0) = 0 \tag{G-32c}$$

3. <u>Symmetry of Solution</u>

Consider figure G-1 with $\phi = \pi$. From equations (G-5) and (G-11)

$$\bar{V}_n = [\bar{e}_x u_n(r) + \bar{e}_r v_n(r) + \bar{e}_\theta w_n(r)]e^X \tag{G-33}$$

$$\bar{V}'_n = [\bar{e}_x u_n(r) - \bar{e}_r v_n(r) - \bar{e}_\theta w_n(r)]e^X e^{in\pi} \tag{G-34}$$

Note that $e^{in\pi} = (-1)^n$, and let the components of $\bar{V}_n$ and $\bar{V}'_n$ be U_n, V_n, W_n and U'_n, V'_n and W'_n. From equations (G-33) and (G-34) the components of $\bar{V}_n$ and $\bar{V}'_n$ have the following relations.

$$U_n = U'_n(-1)^n \tag{G-35}$$

$$V_n = -V'_n(-1)^n \tag{G-36}$$

$$W_n = -W'_n(-1)^n \tag{G-37}$$

Thus the components of a vector with the form of $\bar{V}_n$ must act

either as an even or odd function at the origin, depending upon n. But the functional dependence of U_n on r is contained in $u_n(r)$. Thus $u_n(r)$ must look like an even or odd function. Similarly, v_n and w_n must behave like even or odd functions. In summary, equations (G-35), (G-36) and (G-37) imply the following

$$\underline{n\ even} \quad u_n(r) \quad \text{even symmetry} \tag{G-38a}$$

$$\left.\begin{array}{l} v_n(r) \quad \text{(G-38b)} \\ \\ w_n(r) \quad \text{(G-38c)} \end{array}\right\} \text{odd symmetry}$$

$$\underline{n\ odd} \quad u_n(r) \quad \text{odd symmetry} \tag{G-39a}$$

$$\left.\begin{array}{l} v_n(r) \quad \text{(G-39b)} \\ \\ w_n(r) \quad \text{(G-39c)} \end{array}\right\} \text{even symmetry}$$

4. Summary

If the curl operation is performed upon a vector with the conditions of symmetry from part 3, the resulting vector obeys the same rules of symmetry, that is, the specification that even or odd derivatives of the components equal zero is not changed by applying the curl. All vector functions are assumed to be of the form of equation (G-7).

The condition of single-valuedness at the origin for n=0 and n=1 has the same property. Using the boundary conditions at the origin from equations (G-30) or (G-31) and applying the curl operator does not result in different

conditions on the results. This can be verified by performing the curl operation on a general vector of the form of (G-7), imposing the boundary conditions on the original vector's components and observing the resulting restrictions on the components of the curl.

For $n>1$, this is no longer true. Only the cases of $n=0$ and $n=1$ are considered in this paper. Thus, for $n=0$ and $n=1$, the vorticity, velocity and velocity vector potential will all have the boundary conditions previously discussed. The boundary conditions on the velocity vector potential at $r=0$ are summarized below.

$n=0$ all odd derivatives of $f = 0$
all even derivatives of $g = 0$
(including $g(0) = 0$)
all even derivatives of $h = 0$
(including $h(0) = 0$) (G-40)

$n=1$ all even derivatives of $f = 0$
(including $f(0) = 0$)
all odd derivatives of $g = 0$
all odd derivatives of $h = 0$
$g(0) = ih(0)$ (G-41)

```
C.....................................................................
C PRCGRAM #1
C
C                     PROGRAM TO PRINT EIGENVALUES
C                  FOR THE 3-D POISEUILLE FLOW PROBLEM
C
C
C
C.....................................................................
C
C     THIS PROGRAM SOLVES THE LINEARIZED NAVIER STOKES
C     EQUATICN FOR POISEUILLE FLOW.  THE EIGENVALUES
C     RESULTING FROM THE FINITE DIFFERENCE APPROXIMATION
C     ARE PRINTED.
C
C     INPUT
C
C         THE FOLLOWING MAY BE INPUT TO THE PROGRAM AS DATA
C         USING NAMELIST, 'LIST'.  NOTE, THE DEFAULT VALUES
C         ARE CNLY SET INITIALLY AND VALUES SET BY THE      .
C         USER WILL NOT BE CHANGED BETWEEN RUNS
C
C         N - HALF OF THE NUMBER OF FINITE DIFFERENCE GRID
C         PCINTS ACROSS THE CHANNEL NOT INCLUDING THE END
C         POINTS.  N MUST BE .LE. MDIM, WHICH IS THE
C         DIMENSION OF THE MATRICES IN THIS PROGRAM.
C         DEFAULTED TO THE VALUE OF NDIM, THAT IS THE
C         DIMENSION OF THE MATRICES.  SEE PROGRAM BELOW FOR
C         THE DEFAULT VALUE.
C
C         REY - THE REYNOLDS NUMBER  (REAL*8)  DEFAULT
C         VALUE = 6000.0
C
C         AR,AI,BR,BI - THE REAL AND IMAGINARY PARTS OF THE
C         PERTURBATION WAVE NUMBERS  (REAL*8)
C         DEFAULTED TO 0.0, 1.0, 0.0, AND 0.0 RESPECTIVELY
C
C         VEL - THE VELOCITY OF THE MOVING COORDINATE
C         REFERENCE SYSTEM FOR WHICH THE STABILITY IS
C         DETERMINED.  (REAL*8)  DEFAULTED TO 0.0
C
C     CUTPUT
C
C         THE CUTPUT OF THIS PROGRAM IS A TABULATION OF THE
C         EIGENVALUES.  TWO LISTS ARE PRINTED, ONE FOR THE
C         EIGENVALUES CORRESPONDING TO EVEN EIGENFUNCTIONS
C         ANC CNE FOR THOSE CORRESPONDING TO ODD EIGEN-
C         FUNCTICNS.  THE STABILITY OF EACH EIGENVALUE IS
C         PRINTED AND THE LEAST STABLE EIGENVALUE IS MARKED
C         WITH ASTERISKS.  A PLOT OF THE EIGENVALUES IS ALSO
C         PRINTED.
C
C     SUBROUTINES
C
C         THIS PROGRAM CALLS THE SUBROUTINE 'DEIGEO' TO
C         SOLVE FOR THE EIGENVALUES.  SUBROUTINE 'PLOTP'
C         IS USED TO PLOT THE EIGENVALUES ON THE PRINTER.
C
C
C
C.....................................................................
C
      IMPLICIT REAL*8 (A-H,O-Z)
      COMPLEX*16 A,B
      REAL*4 GR4(60),GI4(60)
      REAL*8 GRE(60),GIE(60),GRO(60),GIO(60)
      COMPLEX*16 XMAT(60,30,3)
         COMPLEX*16 YMAT(60,30),WVEC(60),BMAT(5,60)
         EQUIVALENCE(YMAT(1,1),XMAT(1,1,3)),
     *               (BMAT(1,1),XMAT(1,1,3)),
     *               (WVEC(1),XMAT(1,6,3))
      NAMELIST / LIST / N,REY,AR,AI,BR,BI,VEL
C
C INITIALIZE VARIABLES  (SET DEFAULT VALUES)
```

```
C
      MDIM = 60
      N = 60
      REY = 6000D0
      AR = 0D0
      AI = 1D0
      BR = 0D0
      BI = 0D0
C
C READ NAMELIST AND SET ALPHA AND BETA
C
    1 READ(5,LIST,END=100)
      A = DCMPLX(AR,AI)
      B = DCMPLX(BR,BI)
C
C PRINT INPUT VALUES AS PAGE HEADING FOR EIGENVALUE LIST
C
      WRITE(6,9004)
 9004 FORMAT('1')
      WRITE(6,9005) N,REY,A,B,VEL
 9005 FORMAT(' N =',I4,/,' REY =',F10.2,8X,'ALPHA =',
     *      2F12.7,8X,'BETA =',2F12.7,/,' VEL =',F7.2)
C
C CALL SUBROUTINE TO SOLVE FOR EIGENVALUES.
C
      CALL DEIGEO(A,B,REY,N,MDIM,GRE,GIE,GRO,GIO,XMAT,YMAT,
     *      BMAT,WVEC)
C
C DETERMINE WHICH EIGENVALUE IS THE LEAST STABLE.
C
      TEMP = -1D10
      MARK = 1
C
      DO 20 I=1,N
      IF(GRO(I)+AR*VEL.LT.TEMP) GO TO 20
      TEMP = GRO(I)+AR*VEL
      ITEMP = I
   20 CONTINUE
C
      DO 40 I=1,N
      IF(GRE(I)+AR*VEL.LT.TEMP) GO TO 40
      TEMP = GRE(I)+AR*VEL
      ITEMP = I
      MARK = 2
   40 CONTINUE
C
C LIST EIGENVALUES FOR ODD EIGENFUNCTIONS
C
      WRITE(6,9003)
 9003 FORMAT(///,6X,'GAMMA REAL',5X,'GAMMA IMAG',12X,'STAB')
      WRITE(6,9006)
 9006 FORMAT('0EIGENVALUES FOR ODD EIGENVECTORS',/)
C
      DO 50 I=1,N
      TEMP = GRO(I)+AR*VEL
      WRITE(6,9000) GRO(I),GIO(I),TEMP
      IF(I.EQ.ITEMP.AND.MARK.EQ.1) WRITE(6,9001)
   50 CONTINUE
 9000 FORMAT('0',1P2D15.4,1PD20.4)
 9001 FORMAT('+',52X,'***')
C
C LIST EIGENVALUES FOR EVEN EIGENVECTORS.
C
      WRITE(6,9007)
 9007 FORMAT('0EIGENVALUES FOR EVEN EIGENVECTORS',/)
      DO 55 I=1,N
      TEMP = GRE(I)+AR*VEL
      WRITE(6,9000) GRE(I),GIE(I),TEMP
      IF(I.EQ.ITEMP.AND.MARK.EQ.2) WRITE(6,9001)
   55 CONTINUE
C
C PUT EIGENVALUES INTO SINGLE PRECISION VECTORS TO PASS TO
```

```
C SUBROUTINE TO DO PLOTTING FOR ODD FUNCTIONS.
C
      DO 60 I=1,N
      GR4(I) = SNGL(GRO(I))
   60 GI4(I) = SNGL(GIO(I))
      WRITE(6,9004)
      CALL PLOTP(GR4,GI4,N,0)
      WRITE(6,9005) N,REY,A,B,VEL
C
C SIMILARLY PLOT EIGENVALUES FOR EVEN EIGENFUNCTIONS.
C
      DO 65 I=1,N
      GR4(I) = SNGL(GRE(I))
   65 GI4(I) = SNGL(GIE(I))
      WRITE(6,9004)
      CALL PLOTP(GR4,GI4,N,0)
      WRITE(6,9005) N,REY,A,B,VEL
C
      GO TO 1
  100 WRITE(6,9004)
      STOP
      END
```

```
C.................................................................
C PRCGRAM #1A
C
C                      PROGRAM TO PRINT EIGENVALUES
C                   FOR THE 3-D POISEUILLE FLOW PROBLEM
C                      AND PLOT CN CALCCMP PLOTTER
C
C
C
C.................................................................
C
C     THIS PRCGRAM SOLVES THE LINEARIZED NAVIER STOKES
C     EQUATICN FOR POISEUILLE FLOW.  THE EIGENVALUES
C     RESULTING FROM THE FINITE DIFFERENCE APPROXIMATION
C     ARE PRINTED.
C
C     INPUT
C
C         THE FCLLOWING MUST BE INPUT TO THE PROGRAM AS DATA
C         USING NAMELIST, 'LIST'.
C
C         N - HALF OF THE NUMBER OF FINITE DIFFERENCE GRID
C         POINTS ACROSS THE CHANNEL NOT INCLUDING THE END
C         POINTS.  N MUST BE .LE. MDIM, WHICH IS THE
C         DIMENSION OF THE MATRICES IN THIS PROGRAM.
C         DEFAULTED TO THE VALUE OF MDIM, THAT IS, THE
C         DIMENSION OF THE MATRICES.  SEE PROGRAM BELOW FOR
C         THE DEFAULT VALUES.
C
C         REY - THE REYNOLDS NUMBER  (REAL*8)
C         DEFAULT VALUE = 6000.
C
C         AR,AI,BR,BI - THE REAL AND IMAGINARY PARTS OF THE
C         PERTURBATION WAVE NUMBERS  (REAL*8).  DEFAULTED
C         TO 0., 1., 0. AND 0. RESPECTIVELY.
C
C         VEL - THE VELOCITY OF THE MOVING COORDINATE
C         REFERENCE SYSTEM FOR WHICH THE STABILITY IS
C         DETERMINED.  (REAL*8).  DEFAULTED TO 0.
C
C         GRSCL,GISCL - THE X AND Y SCALE OF THE OUTPUT
C         GRAPH IN UNITS PER INCH.  BOTH DEFAULTED TO 0.2.
C         SEE SUBROUTINE DRAW FOR RESTRICTIONS ON THEIR
C         VALUES
C
C         NW,NH - THE WIDTH AND HEIGHT OF THE OUTPUT GRAPH
C         IN INCHES.  DEFAULTED TO 6 AND 8 RESPECTIVELY.
C         SEE SUBROUTINE DRAW FOR RESTRICTIONS ON THEIR
C         VALUES
C
C         TITLE - A VECTOR OF LENGTH 12 WHICH IS PRINTED
C         WITH THE GRAPH (REAL*8)
C
C     OUTPUT
C
C         THE OUTPUT OF THIS PROGRAM IS A TABULATION OF THE
C         EIGENVALUES.  TWO LISTS ARE PRINTED, ONE FOR THE
C         EIGENVALUES CORRESPONDING TO EVEN EIGENFUNCTIONS
C         AND CNE FOR THOSE CORRESPONDING TO ODD EIGEN-
C         FUNCTICNS.  THE STABILITY OF EACH EIGENVALUE IS
C         PRINTED AND THE LEAST STABLE EIGENVALUE IS MARKED
C         WITH ASTERISKS.  THE EIGENVALUES ARE GRAPHED BY
C         THE CALCCMP PLOTTER.
C
C     SUBROUTINES
C
C         THIS PROGRAM CALLS THE SUBROUTINE 'DEIGEO' TO
C         SOLVE FOR THE EIGENVALUES AND SUBROUTINE 'DRAW'
C         TO GRAPH THE RESULTS.
C
C
C
C.................................................................
C
```

```
      IMPLICIT REAL*8 (A-H,O-Z)
      COMPLEX*16 A,B
      REAL*4 GR4(30),GI4(30)
      REAL*8 GRE(60),GIE(60),GRO(60),GIO(60)
      COMPLEX*16 XMAT(60,30,3)
         COMPLEX*16 YMAT(60,30),WVEC(60),BMAT(5,60)
         EQUIVALENCE(YMAT(1,1),XMAT(1,1,3)),
     *              (BMAT(1,1),XMAT(1,1,3)),
     *              (WVEC(1),XMAT(1,6,3))
      REAL*8 LABEL,TITLE(12)
      REAL*4 GRSCL,GISCL
      DATA LABEL/'        '/
      DATA TITLE/'HARRISON',' 0279   ',10*'        '/
      NAMELIST / LIST / N,REY,AR,AI,BR,BI,VEL,GRSCL,GISCL,
     *     NW,NH,TITLE
C
C INITIALIZE VARIABLES (SET DEFAULT VALUES)
C
      MDIM = 60
      N = 60
      REY = 6000D0
      AR = 0D0
      AI = 1D0
      BR = 0D0
      BI = 0D0
      VEL = 0D0
      GRSCL = .20
      GISCL = .20
      NW = 6
      NH = 8
C
C READ NAMELIST AND SET ALPHA AND BETA
C
    1 READ(5,LIST,END=100)
      A = DCMPLX(AR,AI)
      B = DCMPLX(BR,BI)
C
C PRINT INPUT VALUES AS PAGE HEADING FOR EIGENVALUE LIST.
C
      WRITE(6,9004)
 9004 FORMAT('1')
      WRITE(6,9005) N,REY,A,B,VEL
 9005 FORMAT(' N =',I4,/,' REY =',F10.2,8X,'ALPHA =',
     *     2F12.7,8X,'BETA =',2F12.7,/,' VEL =',F7.2)
C
C CALL SUBROUTINE TO SOLVE FOR EIGENVALUES
C
      CALL DEIGEO(A,B,REY,N,MDIM,GRE,GIE,GRO,GIO,XMAT,YMAT,
     *     BMAT,WVEC)
C
C DETERMINE LEAST STABLE EIGENVALUE
C
      TEMP = -1D10
      MARK = 1
C
      DO 20 I=1,N
      IF(GRO(I)+AR*VEL.LT.TEMP) GO TO 20
      TEMP = GRO(I)+AR*VEL
      ITEMP = I
   20 CONTINUE
C
      DO 40 I=1,N
      IF(GRE(I)+AR*VEL.LT.TEMP) GO TO 40
      TEMP = GRE(I)+AR*VEL
      ITEMP = I
      MARK = 2
   40 CONTINUE
C
C LIST EIGENVALUES FOR ODD EIGENVECTORS
C
      WRITE(6,9003)
 9003 FORMAT(///,6X,'GAMMA REAL',5X,'GAMMA IMAG',12X,'STAB')
```

```
      WRITE(6,9006)
 9006 FORMAT('OEIGENVALUES FOR ODD EIGENVECTORS',/)
C
      DO 50 I=1,N
      TEMP = GRO(I)+AR*VEL
      WRITE(6,9000) GRO(I),GIO(I),TEMP
      IF(I.EQ.ITEMP.AND.MARK.EQ.1) WRITE(6,9001)
   50 CONTINUE
 9000 FORMAT('0',1P2D15.4,1PD20.4)
 9001 FORMAT('+',52X,'***')
C
C LIST EIGENVALUES FOR EVEN EIGENVECTORS.
C
      WRITE(6,9007)
 9007 FORMAT('OEIGENVALUES FOR EVEN EIGENVECTORS',/)
      DO 55 I=1,N
      TEMP = GRE(I)+AR*VEL
      WRITE(6,9000) GRE(I),GIE(I),TEMP
      IF(I.EQ.ITEMP.AND.MARK.EQ.2) WRITE(6,9001)
   55 CONTINUE
C
C PUT EIGENVALUES INTO REAL*4 VECTOR IN GROUPS OF 30 OR LESS
C TO PASS TO SUBROUTINE DRAW.  CALL DRAW REPEATEDLY UNTIL
C ALL EIGENVALUES FOR ODD EIGENVECTORS ARE PLOTTED.  DO THE
C SAME FOR THE EIGENVALUES FOR EVEN EIGENVECTORS.
C
      NGCODE = 1
      IE = 30
      IMODE = 0
      IF(30.GT.N) IE = N
C
   60 DO 65 I=1,IE
      GR4(I) = SNGL(GRO(IMODE*30+I))
   65 GI4(I) = SNGL(GIO(IMODE*30+I))
C
      CALL DRAW(IE,GR4,GI4,NGCODE,3,LABEL,TITLE,
     *    GRSCL,GISCL,NH,NW,2,2,NW,NH,0,LAST)
C
      NGCODE = 2
      IF((IMODE+1)*30+1.GT.N) NGCODE = 3
      DO 70 I=1,IE
      GR4(I) = SNGL(GRE(IMODE*30+I))
   70 GI4(I) = SNGL(GIE(IMODE*30+I))
C
      CALL DRAW(IE,GR4,GI4,NGCODE,1,LABEL,TITLE,
     *    GRSCL,GISCL,NH,NW,2,2,NW,NH,0,LAST)
C
      IF(NGCODE.EQ.3) GO TO 80
      IMODE = IMODE + 1
      IF((IMODE+1)*30.GT.N) IE = N - IMODE*30
      GO TO 60
C
   80 GO TO 1
  100 STOP
      END
```

```
C.......................................................................
C PROGRAM #2
C
C                  PROGRAM TO FIND THE SHAPE OF
C                      THE STABILITY CURVES
C                    FOR 3-D POISEUILLE FLOW
C
C.......................................................................
C
C     THIS PROGRAM FINDS THE SHAPE OF THE STABILITY CURVES.
C     IT PERFORMS FIVE FUNCTIONS DEPENDING ON THE VALUE OF
C     THE INPUT PARAMETER 'MODE'.  INPUT IS BY THE
C     NAMELIST 'LIST'.  OTHER INPUT AND OUTPUT ARE
C     DETERMINED BY THE VALUE OF MODE AS WELL AS THE
C     FUNCTION PERFORMED.  IN ALL CASES, ONLY THE PARAMETERS
C     AI AND REY ARE VARIED.  OTHER PARAMETERS ARE ALWAYS
C     HELD CONSTANT THROUGHOUT THE CALCULATIONS.
C
C     MODE = 1 .........................................
C
C     THE STABILITY AT ANY GIVEN POINT IS FOUND
C
C     INPUT
C
C         AR,AI,BR,BI - THE REAL AND IMAGINARY PARTS
C         OF ALPHA AND BETA. (REAL*8)
C
C         VEL - THE VELOCITY OF THE MOVING COORDINATE
C         FRAME FOR WHICH THE STABILITY IS TO BE
C         DETERMINED. (REAL*8)
C
C         REY - THE REYNOLDS NUMBER. (REAL*8)
C
C     OUTPUT
C
C         THE PROGRAM PRINTS THE INPUT AND THE RESULTING
C         STABILITY.
C
C     MODE = 2 .........................................
C
C     FOR A GIVEN VALUE OF REY, AI IS ADJUSTED TO FIND THE
C     POINT WHICH HAS THE DESIRED STABILITY.
C
C     INPUT
C
C         AR,BR,BI,REY,VEL - VALUES TO BE HELD CONSTANT
C         DURING THE SEARCH FOR THE VALUE OF AI.
C
C         AI - AN INITIAL GUESS.
C
C         DAI - AN ESTIMATE OF THE ERROR IN THE GUESS
C         FOR AI. (REAL*8)
C
C         STAB - THE STABILITY FOR WHICH THE VALUE OF AI IS
C         DESIRED. (REAL*8)
C
C     OUTPUT
C
C         THE INPUT VALUES INCLUDING THE INITIAL GUESS FOR
C         AI ARE PRINTED ALONG WITH THE FINAL APPROXIMATION
C         FOR AI.
C
C     MODE = 3 .........................................
C
C     FOR A GIVEN VALUE OF AI, REY IS ADJUSTED TO FIND THE
C     POINT WHICH HAS THE DESIRED STABILITY.
C
C     INPUT
C
C         AR,AI,BR,BI - VALUES TO BE HELD CONSTANT DURING
C         THE SEARCH FOR THE VALUE OF REY.
C
```

```
C          REY - AN INITIAL GUESS.
C
C          DREY - AN ESTIMATE OF THE ERROR IN THE GUESS FOR
C          REY. (REAL*8)
C
C          STAB - THE STABILITY FOR WHICH THE VALUE OF REY
C          IS DESIRED.
C
C      OUTPUT
C
C          THE INPUT VALUES INCLUDING THE INITIAL GUESS
C          FOR REY ARE PRINTED ALONG WITH THE FINAL
C          APPROXIMATION FOR REY.
C
C      MODE = 4 ..............................................
C
C      THE POINT OF MINIMUM REY ON THE NEUTRAL STABILITY
C      CURVE IS DETERMINED.
C
C      INPUT
C
C          AR,BR,BI,VEL - VALUES TO BE HELD CONSTANT DURING
C          THE DETERMINATION OF THE MINIMUM REY.
C
C          AI,REY - INITIAL GUESS FOR THE POINT OF
C          MINIMUM REY.
C
C          DAI,DREY - AN ESTIMATE OF THE ERROR IN THE
C          GUESSES FOR AI AND REY.
C
C      OUTPUT
C
C          THE PROGRAM PRINTS THE INITIAL GUESS, THE
C          SUCCESSIVE APPROXIMATIONS AND THE FINAL
C          ESTIMATE FOR THE VALUES OF AI AND REY.
C
C      MODE = 5 ..............................................
C
C      THE POINT OF MAXIMUM INSTABILITY IS DETERMINED.
C
C      INPUT
C
C          AR,BR,BI,VEL - VALUES TO BE HELD CONSTANT
C
C          AI,REY - INITIAL GUESS FOR THE POINT OF MAXIMUM
C          INSTABILITY.
C
C          DAI,DREY - AN ESTIMATE OF THE ERROR IN THE
C          GUESSES FOR AI AND REY.
C
C      OUTPUT
C
C          THE PROGRAM PRINTS THE INITIAL GUESS, THE
C          SUCCESSIVE APPROXIMATIONS AND THE FINAL ESTIMATE
C          FOR THE VALUES OF AI AND REY.
C
C       ..............................................
C
C      SUBROUTINES REQUIRED
C
C          THIS PROGRAM CALLS THE SUBROUTINE 'DFIT2' AND USES
C          THE FUNCTION 'EIGLS'.
C
C
C.....................................................
C
      IMPLICIT REAL*8 (A-H,O-Z)
      NAMELIST / LIST / REY,AR,AI,BI,DAI,DREY,VEL,BR,MODE,
     *   STAB
      WRITE(6,9000)
 9000 FORMAT('1')
C
```

```
C INITIALIZE VARIABLES
C
      STAB = ODO
      VEL=ODO
      BR = ODO
      BI = ODO
      AR = ODO
      AI = 1DO
      DAI = .010DO
      DREY = 250DO
      MODE = 1
C
C READ NAMELIST VARIABLES
C
    1 READ(5,LIST,END=100)
      REY1 = REY
      AITEMP = AI
C
C JUMP TO TYPE OF SOLUTION CALLED FOR BY VALUE OF MODE
C
      GO TO (10,20,30,40,50),MODE
C
C     ..................MODE = 1..............................
C                  FIND STAB FOR SINGLE POINT
C
   10 WRITE (6,9100) REY,AR,AI,BR,BI,VEL
 9100 FORMAT(///,21X,'STABILITY AT POINT...',/,21X,'REY =',
     *      F8.1,/,21X,'AR =',F12.5,5X,'AI =',F12.5,/,21X,
     *      'BR =',F12.5,5X,'BI =',F12.5,/,21X,'FOR VEL =',
     *      F6.2)
C
C USE EIGLS TO FIND STABILITY
C
      G1 = EIGLS(REY,AR,AI,BR,BI,VEL)
      WRITE(6,9101) G1
 9101 FORMAT(/,21X,'STAB =',1PD12.4)
      GO TO 1
C
C     ..................MODE = 2..............................
C                  FIND AI GIVEN REY AND STAB
C
   20 WRITE(6,9200) STAB,REY,AI,BR,BI,AR,VEL,DAI
 9200 FORMAT(///,21X,'FOR STAB =',F9.5,5X,'AT REY =',F8.1,
     *      /,21X,'INITIAL GUESS...  AI =',F12.5,/,21X,
     *      'GIVEN BR =',F12.5,5X,'BI =',F12.5,/,27X,'AR =',
     *      F12.5,5X,'VEL =',F6.2,/,21X,'DAI =',F10.5,/)
C
C CALCULATE STABILITY FOR INITIAL GUESS.
C
      G1 = EIGLS(REY,AR,AI,BR,BI,VEL)
      WRITE(6,9201) AI,G1
 9201 FORMAT(21X,'FOR AI =',F9.5,5X,'STAB =',1PD12.4)
C
C ADD DAI TO INITIAL AI AND DETERMINE STABILITY.
C
      AI1 = AI + DAI
      G2 = EIGLS(REY,AR,AI1,BR,BI,VEL)
      WRITE(6,9201) AI1,G2
C
C USE LINEAR APPROXIMATION TO FIND NEW VALUE OF AI.
C CALCULATE STABILITY.
C
      AI2 = AI1 + (AI1-AI)*(STAB-G2)/(G2-G1)
      G3 = EIGLS(REY,AR,AI2,BR,BI,VEL)
      WRITE(6,9201) AI2,G3
C
C USE SUBROUTINE DFIT2 TO OBTAIN SECOND ORDER APPROXIMATION
C FOR FINAL VALUE OF AI.
C
      CALL DFIT2(AI,AI1,AI2,G1,G2,G3,STAB,AIF,DUMMY,DUMY2)
      WRITE(6,9202) AIF
 9202 FORMAT(/,21X,'FINAL ESTIMATE... AI =',F12.5)
```

```
      GC TO 1
C
C     ..................MODE = 3...............................
C              FIND REY GIVEN AI AND STAB
C
   30 WRITE(6,9300) STAB,AI,REY,BR,BI,AR,VEL,DREY
 9300 FCRMAT(///,21X,'FOR STAB =',F9.5,5X,'AT AI =',F12.5,
     *     /,21X,'INITIAL GUESS...  REY =',F8.1,/,21X,
     *     'GIVEN BR =',F12.5,5X,'BI =',F12.5,/,27X,
     *     'AR =',F12.5,5X,'VEL =',F6.2,/,21X,'DREY =',
     *     F8.1,/)
C
C CALCULATE STABILITY FOR INITIAL GUESS
C
      G1 = EIGLS(REY,AR,AI,BR,BI,VEL)
      WRITE(6,9301) REY,G1
 9301 FORMAT(21X,'FOR REY =',F8.1,5X,'STAB =',1PD12.4)
C
C ACC CREY TC INITIAL REY AND DETERMINE STABILITY.
C
      REY1 = REY + DREY
      G2 = EIGLS(REY1,AR,AI,BR,BI,VEL)
      WRITE(6,9301) REY1,G2
C
C USE LINEAR APPROXIMATION TO FIND NEW VALUE OF REY.
C CALCULATE STABILITY.
C
      REY2 = REY1 + (REY1-REY)*(STAB-G2)/(G2-G1)
      G3 = EIGLS(REY2,AR,AI,BR,BI,VEL)
      WRITE(6,9301) REY2,G3
C
C USE SUBRCUTINE DFIT2 TO OBTAIN SECCND ORDER APPROXIMATION
C FOR FINAL VALUE OF REY.
C
      CALL DFIT2(REY,REY1,REY2,G1,G2,G3,STAB,REYF,DUMMY,
     *     CUMY2)
      WRITE(6,9302) REYF
 9302 FORMAT(/,21X,'FINAL ESTIMATE...  REY =',F8.1)
      GO TO 1
C
C     .....................MODE = 4.........................
C        FINC POINT OF MINIMUM REYNOLDS NUMBER
C             ON THE NEUTRAL STABILITY CURVE
C
C OUTPUT VALUES FOR INITIAL GUESS
C
   40 WRITE(6,9400) AI,REY,DAI,DREY,BI,AR,VEL,BR
 9400 FORMAT('1',////,21X,'POINT OF MINIMUM REY ON NEUTRAL',
     *     ' STABILITY CURVE',/,21X,
     *     'INITIAL GUESS...  AI =',F12.5,5X,
     *     'REY1 =',F8.1,/,21X,'DAI =',F12.5,5X,'DREY =',
     *     F8.1,/,21X,'BI =',F12.5,5X,'AR =',F12.5,
     *     /,21X,'VEL =',F6.2,5X,'BR =',F12.5,///)
C
C FOR INITIAL REYNOLDS NUMBER, CHOOSE TWC MORE VALUES FOR
C AI AND USE PARABOLIC CURVE FIT TO FIND MINIMUM STABILITY
C WITH CORRESPCNCING VALUE FOR AI
C
      AIM = AI-CAI
      AIP = AI+DAI
      G1 = EIGLS(REY1,AR,AIM,BR,BI,VEL)
      WRITE(6,9401) AIM,G1
 9401 FORMAT(21X,'REY1...     AI =',F12.5,5X,'FOR STAB =',
     *     1PD12.4)
      G2 = EIGLS(REY1,AR,AI,BR,BI,VEL)
      WRITE(6,9401) AI,G2
      G3 = EIGLS(REY1,AR,AIP,BR,BI,VEL)
      WRITE(6,9401) AIP,G3
      CALL DFIT2(AIM,AI,AIP,G1,G2,G3,1D2,DUMMY,AI1,GMIN1)
      WRITE(6,9402) AI1,GMIN1
 9402 FORMAT('0',20X,'AI =',F12.5,5X,'FOR MAX G =',
     *     1PC15.4,/)
```

```
C
C CHCCSE NEXT GUESS FOR REYNOLC'S NUMBER USING DREY
C
      REY2 = REY1+DREY
      IF(GMIN1.GT.ODO) REY2 = REY1-DREY
      AI = AI1
      WRITE(6,9403) AI,REY2
 9403 FORMAT(///,21X,'SECOND GUESS...  AI =',F12.5,5X,
     *     'REY2 =',F8.1,/)
C
C FIND MINIMUM STABILITY AND CORRESPONDING AI FOR SECOND
C GUESS OF REY
C
      AIM = AI-DAI
      AIP = AI+DAI
      G1 = EIGLS(REY2,AR,AIM,BR,BI,VEL)
      WRITE(6,9404) AIM,G1
 9404 FORMAT(21X,'REY2...      AI =',F12.5,5X,'FOR STAB =',
     *     1PD12.4)
      G2 = EIGLS(REY2,AR,AI,BR,BI,VEL)
      WRITE(6,9404) AI,G2
      G3 = EIGLS(REY2,AR,AIP,BR,BI,VEL)
      WRITE(6,9404) AIP,G3
      CALL DFIT2(AIM,AI,AIP,G1,G2,G3,1D2,DUMMY,AI2,GMIN2)
      WRITE(6,9402) AI2,GMIN2
C
C USE LINEAR FIT TO DETERMINE THIRD GUESS FOR REY AND AI
C
      REY3 = REY1-(REY1-REY2)*GMIN1/(GMIN1-GMIN2)
      AI = AI1-(AI1-AI2)*GMIN1/(GMIN1-GMIN2)
      WRITE(6,9405) AI,REY3
 9405 FORMAT(///,21X,'THIRD GUESS...  AI =',F12.5,5X,
     *     'REY3 =',F8.1,/)
C
C REPEAT PROCEDURE TO FIND MINIMUM STABILITY AT THIRD GUESS
C FOR REY
C
      AIM = AI-DAI
      AIP = AI+DAI
      G1 = EIGLS(REY3,AR,AIM,BR,BI,VEL)
      WRITE(6,9406) AIM,G1
 9406 FORMAT(21X,'REY3...      AI =',F12.5,5X,'FOR STAB =',
     *     1PD12.4)
      G2 = EIGLS(REY3,AR,AI,BR,BI,VEL)
      WRITE(6,9406) AI,G2
      G3 = EIGLS(REY3,AR,AIP,BR,BI,VEL)
      WRITE(6,9406) AIP,G3
      CALL DFIT2(AIM,AI,AIP,G1,G2,G3,1D2,DUMMY,AI3,GMIN3)
      WRITE(6,9402) AI3,GMIN3
C
C USE PARABOLIC FIT TO DETERMINE FINAL GUESS FOR REY AND AI
C
      CALL DFIT2(AI1,AI2,AI3,GMIN1,GMIN2,GMIN3,ODO,AIF,
     *     DUMMY,DUMY2)
      CALL DFIT2(REY1,REY2,REY3,GMIN1,GMIN2,GMIN3,ODO,REYF,
     *     DUMMY,DUMY2)
      WRITE(6,9407) AIF,REYF
 9407 FORMAT(///,21X,'FINAL ESTIMATE...  AI =',F12.5,5X,
     *     'REY =',F8.1,////////)
      AI = AITEMP
      GO TO 1
C
C     .................MODE = 5..............................
C               FIND POINT OF MINIMUM STABILITY
C            (THAT IS, FOR MAX VALUE OF STAB)
C
   50 WRITE(6,9500) AI,REY,DAI,DREY,BI,AR,VEL,BR
 9500 FORMAT('1',///,20X,'POINT OF MINIMUM STABILITY',/,21X,
     *     'INITIAL GUESS...  AI =',F12.5,5X,'REY1 =',F8.1,
     *     /,21X,'DAI =',F12.5,5X,'DREY =',F8.1,/,21X,
     *     'BI =',F12.5,5X,'AR =',F12.5,/,21X,'VEL =',F6.2,
     *     5X,'BR =',F12.5,///)
```

```
C
      AIM = AI - DAI
      AIP = AI + DAI
      G1 = EIGLS(REY1,AR,AIM,BR,BI,VEL)
      WRITE(6,9501) AIM,G1
 9501 FORMAT(21X,'REY1...     AI =',F12.5,5X,'FOR STAB =',
     *      1PD12.4)
      G2 = EIGLS(REY1,AR,AI,BR,BI,VEL)
      WRITE(6,9501) AI,G2
      G3 = EIGLS(REY1,AR,AIP,BR,BI,VEL)
      WRITE(6,9501) AIP,G3
      CALL DFIT2(AIM,AI,AIP,G1,G2,G3,1D2,DUMMY,AI1,GMIN1)
      WRITE(6,9502) AI1,GMIN1
 9502 FORMAT('0',20X,'AI =',F12.5,5X,'FOR STAB =',1PD15.4,/)
C
      REY2 = REY1 + DREY
      AI = AI1
      WRITE(6,9503) AI,REY2
 9503 FORMAT(///,21X,'SECOND GUESS...  AI =',F12.5,5X,
     *      'REY2 =',F8.1,/)
C
      AIM = AI - DAI
      AIP = AI + DAI
      G1 = EIGLS(REY2,AR,AIM,BR,BI,VEL)
      WRITE(6,9504) AIM,G1
 9504 FORMAT(21X,'REY2...     AI =',F12.5,5X,'FOR STAB =',
     *      1PD12.4)
      G2 = EIGLS(REY2,AR,AI,BR,BI,VEL)
      WRITE(6,9504) AI,G2
      G3 = EIGLS(REY2,AR,AIP,BR,BI,VEL)
      WRITE(6,9504) AIP,G3
      CALL DFIT2(AIM,AI,AIP,G1,G2,G3,1D2,DUMMY,AI2,GMIN2)
      WRITE(6,9502) AI2,GMIN2
C
      REY3 = REY1 - DREY
      IF(GMIN2.GT.GMIN1) REY3 = REY2 + DREY
      AI = AI2 + (AI1-AI2)*(REY3-REY2)/(REY1-REY2)
      WRITE(6,9505) AI,REY3
 9505 FORMAT(///,21X,'THIRD GUESS...  AI =',F12.5,5X,
     *      'REY3 =',F8.1,/)
C
      AIM = AI - DAI
      AIP = AI + DAI
      G1 = EIGLS(REY3,AR,AIM,BR,BI,VEL)
      WRITE(6,9506) AIM,G1
 9506 FORMAT(21X,'REY3...     AI =',F12.5,5X,'FOR STAB =',
     *      1PD12.4)
      G2 = EIGLS(REY3,AR,AI,BR,BI,VEL)
      WRITE(6,9506) AI,G2
      G3 = EIGLS(REY3,AR,AIP,BR,BI,VEL)
      WRITE(6,9506) AIP,G3
      CALL DFIT2(AIM,AI,AIP,G1,G2,G3,1D2,DUMMY,AI3,GMIN3)
      WRITE(6,9502) AI3,GMIN3
C
      CALL DFIT2(REY1,REY2,REY3,GMIN1,GMIN2,GMIN3,1D0,DUMMY,
     *      REYF,GF)
C
      A = ((AI1-AI2)/(REY1-REY2)-(AI1-AI3)/(REY1-REY3))
     *      /((REY1**2-REY2**2)/(REY1-REY2)
     *      -(REY1**2-REY3**2)/(REY1-REY3))
      B = (AI1-AI2-A*(REY1**2-REY2**2))/(REY1-REY2)
      C = AI1-A*REY1**2-B*REY1
      AIF = A*REYF**2+B*REYF+C
C
      WRITE(6,9507) AIF,REYF,GF
 9507 FORMAT(///,21X,'FINAL ESTIMATE...  AI =',F12.5,5X,
     *      'REY =',F8.1,/,41X,'STAB =',1PD15.4,////////)
      GO TO 1
C
  100 WRITE(6,9000)
      STOP
      END
```

```
C..........................FUNCTION EIGLS..................
C
C      PURPOSE
C
C        EIGLS RETURNS THE STABILITY OF THE LEAST STABLE
C        EIGENVALUE FOR PLANE POISEULLE FLOW AS DETERMINED
C        FROM A REFERENCE FRAME MOVING IN THE X DIRECTION
C        WITH VELOCITY VEL.
C
C      USAGE
C
C        STAB = EIGLS(REY,AR,AI,BR,BI,VEL)
C
C      DESCRIPTION OF PARAMETERS
C
C        THE FOLLOWING MUST BE SET BY THE CALLING PROGRAM...
C          REY,AR,AI,BR,BI,VEL
C
C        REY - THE REYNOLDS NUMBER (REAL*8)
C        AR,AI,BR,BI - THE REAL AND IMAGINARY PARTS OF THE
C          PERTURBATION WAVE NUMBERS, ALPHA AND BETA
C          (REAL*8)
C
C        VEL - THE VELOCITY OF THE COORDINATE SYSTEM FOR
C          WHICH THE STABILITY IS DESIRED.  (THE MINIMUM,
C          AVERAGE, AND MAXIMUM VELOCITIES OF THE FLUID ARE
C          0.0, 1.0, AND 1.5 RESPECTIVELY)  (REAL*8)
C
C        THE VALUE OF THE STABILITY OF THE LEAST STABLE
C        EIGENVALUE IS RETURNED IN 'EIGLS'.  EIGLS MUST BE
C        REAL*8 IN THE CALLING PROGRAM.
C
C      OTHER ROUTINES NEEDED
C
C        DEIGEO
C
C...........................................................
C
      FUNCTION EIGLS(REY,AR,AI,BR,BI,VEL)
      IMPLICIT REAL*8 (A-H,O-Z)
      REAL*8 GRE(60),GIE(60),GRO(60),GIO(60)
      COMPLEX*16 XMAT(60,30,3)
         COMPLEX*16 YMAT(60,30),WVEC(60),BMAT(5,60)
         EQUIVALENCE(YMAT(1,1),XMAT(1,1,3)),
     *               (BMAT(1,1),XMAT(1,1,3)),
     *               (WVEC(1),XMAT(1,6,3))
C
      MDIM = 60
      N = 60
C
C CALL SUBROUTINE TO DETERMINE EIGENVALUES FOR BOTH EVEN
C AND ODD EIGENFUNCTIONS.
C
      CALL DEIGEO(DCMPLX(AR,AI),DCMPLX(BR,BI),REY,N,MDIM,
     *     GRE,GIE,GRO,GIO,XMAT,YMAT,BMAT,WVEC)
C
C DETERMINE STABILITY FOR EACH EIGENVALUE SAVING THE VALUE
C FOR THE LEAST STABLE ONE.
C
      TEMP = -1D10
      DO 10 I=1,N
      IF(GRE(I)+AR*VEL.LT.TEMP) GO TO 5
      TEMP = GRE(I)+AR*VEL
    5 IF(GRO(I)+AR*VEL.LT.TEMP) GO TO 10
      TEMP = GRO(I)+AR*VEL
   10 CONTINUE
C
C RETURN THE VALUE OF THE MINIMUM STABILITY IN 'EIGLS'.
C
      EIGLS = TEMP
      RETURN
      END
```

```
C...............FUNCTION CHM1E1 AND CHM2E1..................
C                   (CARTESIAN COORDINATES)
C
C
C      PURPOSE
C
C         CHM1E1 AND CHM2E1 RETURN THE VALUES (COMPLEX*16) OF
C         THE COEFFICIENTS FOR THE MATRICES IN THE FINITE
C         DIFFERENCE FORM OF THE LINEARIZED NAVIER-STOKES
C         EQUATION FOR POISEUILLE FLOW.  BOTH FUNCTIONS RESULT
C         FROM THE LINEAR COMBINATION OF EQUATION 1 AND
C         EQUATION 3 TO ELIMINATE THE VELOCITY VECTOR
C         POTENTIAL COMPONENT G AND ARBITRARILY SETTING THE
C         COMPONENT F TO ZERO.  SO, THEY ARE THE COEFFICIENTS
C         FOR THE VECTOR POTENTIAL COMPONENT H.  CHM2E1
C         RETURNS THE TERMS WHICH ARE COEFFICIENTS OF THE
C         EIGENVALUE, GAMMA, AND CHM1E1 RETURNS THE REMAINING
C         TERMS.
C
C      USAGE
C
C         X1 = CHM1E1(K,Y)
C         X2 = CHM2E1(K,Y)
C
C            (CHM1E1 AND CHM2E1 MUST BE DECLARED COMPLEX*16 IN
C            CALLING PROGRAM)
C
C      DESCRIPTION OF PARAMETERS
C
C         THE FOLLOWING PARAMETERS MUST BE SET BY THE CALLING
C            PROGRAM
C
C         K - INDICATES THE POINT ON THE FINITE DIFFERENCE
C            MESH RELATIVE TO THE CENTRAL POINT IN THE CENTRAL
C            DIFFERENCING SCHEME.  IF THE DIFFERENCE IS BEING
C            FORMED ABOUT THE N-TH POINT THEN K=1 REFERS TO
C            THE POINT N-2, K=2 REFERS TO THE POINT N-1,
C            K=3 REFERS TO N, K=4 REFERS TO N+1, AND K=5
C            REFERS TO N+2.
C         K - INDICATES WHICH POINT ON THE FINITE DIFFERENCE
C            MESH IS REFERRED TO THE CENTRAL POINT.  IF THE
C            DIFFERENCE IS BEING FORMED ABOUT THE N-TH POINT
C            THEN K=1 REFERS TO THE POINT N-2, K=2 REFERS TO
C            THE POINT N-1, K=3 REFERS TO N, K=4 REFERS TO N+1,
C            AND K=5 REFERS TO N+2.
C
C         Y - THE VALUE OF THE POSITION RELATIVE TO THE CENTER
C            OF THE CHANNEL.  THE TWO BOUNDARIES ARE AT Y=+1
C            AND Y=-1.
C
C      OTHER ROUTINES NEEDED
C
C         NONE
C
C
C...........................................................
C
      FUNCTION CHM1E1(K,Y)
      IMPLICIT COMPLEX*16 (A-H,O-Z)
      COMMON / COEFNT / A,B,G,REY,DEL
      REAL*8 REY,Y,DEL
C
C THE FOLLOWING FUNCTIONS (M1) EVALUATE THE COEFFICIENTS
C OF THE DERIVATIVES OF H FOR ALL TERMS EXCEPT THOSE
C CONTAINING GAMMA.
C
      CH4M1(Y) = A/REY
      CH2M1(Y) = -1.5D0*A**2*(1D0-Y**2)+2D0*A*(A**2+B**2)
     *     /REY
      CH0M1(Y) = -A*((A**2+B**2)*(1.5D0*A*(1D0-Y**2)-(A**2
     *     +B**2)/REY)+3D0*A)
C
```

```
C THE REMAINING FUNCTIONS (M2) EVALUATE THE COEFFICIENTS
C OF THE DERIVATIVES OF H WHICH ARE ALSO COEFFICIENTS
C OF THE EIGENVALUE GAMMA.
C
      CH2M2(Y) = A
      CHOM2(Y) = A*(A**2+B**2)
C
C SET UP THE FINITE DIFFERENCE VALUES FOR INDEX K FOR M1
C
      GO TO (1,2,3,2,1),K
    1 CHM1E1 = CH4M1(Y)/DEL**4
      GO TO 100
    2 CHM1E1 = -4D0*CH4M1(Y)/DEL**4+CH2M1(Y)/DEL**2
      GO TO 100
    3 CHM1E1 = 6D0*CH4M1(Y)/DEL**4-2D0*CH2M1(Y)/DEL**2
     *     +CHOM1(Y)
  100 RETURN
C
C SET UP THE FINITE DIFFERENCE VALUES FOR INDEX K FOR M2
C
      ENTRY CHM2E1(K,Y)
      GO TO (11,12,13,12,11),K
   11 CHM2E1 = (0D0,0D0)
      GO TO 200
   12 CHM2E1 = CH2M2(Y)/DEL**2
      GO TO 200
   13 CHM2E1 = -2D0*CH2M2(Y)/DEL**2+CHOM2(Y)
  200 RETURN
      END
```

```
C.......................SUBROUTINE MULDBM...................
C
C
C     PURPOSE
C
C       MULDBM PERFORMS THE MATRIX MULTIPLICATION BETWEEN A
C       SQUARE MATRIX X AND A BANDED MATRIX XB WHICH HAS
C       BEEN SET UP BY SUBROUTINE BMSET.  THE RESULT IS
C       PLACED BACK IN X.  THAT IS...
C                    X = (X)(XB)
C
C     USAGE
C
C       CALL MULDBM(X,XB,N,NB,MDIM,TEMPV)
C
C     DESCRIPTION OF PARAMETERS
C
C       THE FOLLOWING MUST BE SET BY THE CALLING PROGRAM.
C         N,MDIM,NB,X,XB,TEMPV
C
C       N - THE SIZE OF THE MATRICES.
C
C       MDIM - THE DIMENSIONS OF THE MATRICES IN THE CALLING
C         PROGRAM.
C
C       NB - THE NUMBER OF BANDS IN THE BANDED MATRIX, XB.
C
C       X - THE SQUARE N BY N MATRIX.  DIMENSIONED
C         (MDIM,MDIM) IN THE CALLING PROGRAM  (COMPLEX*16)
C
C       XB - THE N BY N MATRIX WHICH IS STORED IN BANDED
C         FORM.  DIMENSIONED (NB,MDIM) IN CALLING PROGRAM.
C         (COMPLEX*16)
C
C       TEMPV - A VECTOR WORKSPACE WHICH MUST BE PROVIDED BY
C         THE CALLING PROGRAM.  DIMENSIONED AT LEAST (N).
C         (COMPLEX*16)
C
C       THE FOLLOWING IS OUTPUT BY MULDBM.
C
C       X - SET TO THE PRODUCT OF X AND XB (MDIM,MDIM)
C         (COMPLEX*16)
C
C     REQUIRED ROUTINES
C
C       NONE
C
C
C.............................................................
C
      SUBROUTINE MULDBM(X,XB,N,NB,MDIM,TEMPV)
      COMPLEX*16 X(MDIM,MDIM),XB(NB,MDIM),TEMPV(MDIM),TEMP
      NBHM = (NB-1)/2
      NBHP = (NB+1)/2
C
C LOOP OVER INDEX I
C
      DO 100 I=1,N
C
C STORE COLUMN I OF MATRIX X TEMPORARILY
C
      DO 10 J=1,N
   10 TEMPV(J) = X(I,J)
C
C FIND PRODUCTS FOR FIRST NBHM SPECIAL CASES, THAT IS WHERE
C WHERE THE BANDED MATRIX DOES NOT HAVE ITS FULL WIDTH
C
      DO 22 J=1,NBHM
      TEMP = (0D0,0D0)
      JJ = NBHM + J
      DO 21 K=1,JJ
      JJJ = JJ-K+1
```

```
   21 TEMP = TEMP+TEMPV(K)*XB(JJJ,K)
   22 X(I,J) = TEMP
C
C COMPUTE PRODUCTS FOR "REGULAR" COMBINATIONS OF ROWS AND
C COLUMNS, THAT IS, THOSE THAT ARE NOT TRUNCATED
C AT THE BEGINNING OR END BY THE BOUNDARIES
C
      JF = N-NBHM
      DO 32 J=NBHP,JF
      TEMP = (0D0,0D0)
      DO 31 K=1,NB
      JJJ = NB-K+1
   31 TEMP = TEMP+TEMPV(J-NBHP+K)*XB(JJJ,J-NBHP+K)
   32 X(I,J) = TEMP
C
C FIND PRODUCTS FOR LAST NBHM SPECIAL CASES.
C
      DO 42 J=1,NBHM
      TEMP = (0D0,0D0)
      JJ = NB-J
      DO 41 K=1,JJ
   41 TEMP = TEMP+TEMPV(N-JJ+K)*XB(NB-K+1,N-JJ+K)
   42 X(I,N-NBHM+J) = TEMP
C
  100 CONTINUE
C
      RETURN
      END
```

```
C.....................SUBROUTINE MSET.....................
C
C
C     PURPOSE
C
C       MSET SETS UP A MATRIX FOR THE FINITE DIFFERENCE
C       PROBLEM OF POISEULLE FLOW WITH THE PROPER BOUNDARY
C       CONDITIONS FOR THE VELOCITY VECTOR POTENTIAL FOR
C       VISCOUS FLOW.
C
C     USAGE
C
C       CALL MSET(X,N,MDIM,MODE,CFMAT)
C
C     DESCRIPTION OF PARAMETERS
C
C       THE FOLLOWING MUST BE SET BY THE CALLING PROGRAM
C         N,MDIM,MODE,CFMAT
C
C       N - THE SIZE OF THE MATRIX.  IF MODE=0, N IS EQUAL
C         TO THE NUMBER OF POINTS OF THE FINITE DIFFERENCE
C         MESH ACROSS THE CHANNEL NOT INCLUDING THE TWO
C         BOUNDARIES.  IF MODE=1 OR MODE=2, N IS EQUAL TO
C         ONE-HALF OF THE NUMBER OF INNER POINTS ACROSS THE
C         FULL CHANNEL.  IN THIS CASE, THE CHANNEL MUST BE
C         DIVIDED INTO AN EVEN NUMBER OF POINTS SO THAT N
C         WILL BE AN INTEGER.
C
C       MDIM - THE COLUMN DIMENSION OF THE MATRIX X.  MDIM
C         MUST BE .GE. N.
C
C       MODE - IF MODE=0, THE MATRIX IS SET UP FOR THE FULL
C         CHANNEL.  IF MODE=1 OR MODE=2, THE MATRIX IS SET
C         UP FOR THE HALF CHANNEL AND THE BOUNDARY
C         CONDITIONS ARE SET SUCH THAT THE ODD OR EVEN
C         EIGENFUNCTIONS, RESPECTIVELY, ARE SOLVED FOR.
C
C       CFMAT - THE NAME OF A FUNCTION SUBPROGRAM WITH TWO
C         PARAMETERS, K AND Y, INDICATING WHICH TERM OF THE
C         FINITE DIFFERENCING IS DESIRED, AND THE POSITION
C         RELATIVE TO THE CENTER OF THE CHANNEL.  MUST BE
C         DECLARED EXTERNAL IN THE CALLING PROGRAM.
C         (COMPLEX*16)
C
C       THE FOLLOWING IS OUTPUT BY MSET
C
C       X - THE N BY N MATRIX INTO WHICH THE COEFFICIENTS
C         OF THE FINITE DIFFERENCING ARE PUT.  MUST BE
C         DIMENSIONED (MDIM,MDIM) IN THE CALLING PROGRAM.
C         (COMPLEX*16)
C
C     NOTES...
C
C       THE EIGENVALUES AND VECTORS OBTAINED BY USING MSET
C         TWICE, WITH MODE=1 AND MODE=2, ARE THE
C         SAME AS USING MSET ONCE WITH MODE=0, BUT WITH
C         N TWICE AS LARGE.
C
C     OTHER ROUTINES NEEDED
C
C       NONE - EXCEPT THE FUNCTION SUBPROGRAM NAME PASSED IN
C       THE PARAMETER 'CFMAT'.
C
C
C..........................................................
C
      SUBROUTINE MSET(X,N,MDIM,MODE,CFMAT)
      REAL*8 REY,Y,DELY,DFLOAT
      COMPLEX*16 A,B,G
      COMMON / COEFNT / A,B,G,REY,DELY
      COMPLEX*16  CFMAT
      COMPLEX*16 X(MDIM,MDIM)
C
```

```
C COMPUTE GRID SIZE FOR FINITE DIFFERENCE MESH ACROSS
C HALF CHANNEL OR FULL CHANNEL
C
      DELY = 2D0/DFLOAT(N+1)
      IF(MODE.EQ.1.OR.MODE.EQ.2) DELY = 2D0/DFLOAT(2*N+1)
C
C CHECK IF MATRIX DIMENSIONED LARGE ENOUGH
C
      IF(N.LE.MDIM) GO TO 1
      WRITE(6,9000)
 9000 FORMAT('0* * * ERROR - ARRAYS NOT DIMENSIONED LARGE',
     *      ' ENOUGH * * *')
      STOP
C
C ZERO ENTIRE MATRIX
C
    1 DO 10 I=1,N
      DO 10 J=1,N
   10 X(I,J) = (0D0,0D0)
C
C DO SPECIAL CASE AT DISTANCE DELY FROM CHANNEL WALL
C INCLUDING BOUNDARY CONDITIONS
C
      Y = 1D0-DELY
      X(1,1) = CFMAT(3,Y)+CFMAT(1,Y)
      X(1,2) = CFMAT(4,Y)
      X(1,3) = CFMAT(5,Y)
C
C DO SPECIAL CASE AT DISTANCE 2*DELY FROM CHANNEL WALL
C INCLUDING BOUNDARY CONDITIONS
C
      Y = 1D0-2D0*DELY
      X(2,1) = CFMAT(2,Y)
      X(2,2) = CFMAT(3,Y)
      X(2,3) = CFMAT(4,Y)
      X(2,4) = CFMAT(5,Y)
C
C DO ALL REGULAR POINTS IN BETWEEN, THAT IS, THOSE VALUES
C OF Y FOR WHICH ALL 5 FINITE DIFFERENCE GRID POINTS ARE
C INTERIOR TO THE CHANNEL
C
      IL = N-2
      DO 20 I=3,IL
      K = I-3
      Y = 1D0-DELY*DFLOAT(I)
      DO 20 J=1,5
   20 X(I,K+J) = CFMAT(J,Y)
C
C FINALLY DO THE TWO SPECIAL CASES WHICH OCCUR EITHER AT
C THE CENTER OF THE CHANNEL OR AT THE OTHER WALL, DEPENDING
C ON THE VALUE OF MODE.  BOUNDARY CONDITIONS ARE SET UP
C DEPENDING ON MODE
C
      Y = 1D0-DELY*DFLOAT(N-1)
      X(N-1,N-3) = CFMAT(1,Y)
      X(N-1,N-2) = CFMAT(2,Y)
      X(N-1,N-1) = CFMAT(3,Y)
      X(N-1,N) = CFMAT(4,Y)
      IF(MODE.EQ.1) X(N-1,N) = CFMAT(4,Y)-CFMAT(5,Y)
      IF(MODE.EQ.2) X(N-1,N) = CFMAT(4,Y)+CFMAT(5,Y)
C
      Y = 1D0-DELY*DFLOAT(N)
      X(N,N-2) = CFMAT(1,Y)
      X(N,N-1) = CFMAT(2,Y)
      IF(MODE.EQ.1) X(N,N-1) = CFMAT(2,Y)-CFMAT(5,Y)
      IF(MODE.EQ.2) X(N,N-1) = CFMAT(2,Y)+CFMAT(5,Y)
      X(N,N) = CFMAT(3,Y)+CFMAT(5,Y)
      IF(MODE.EQ.1) X(N,N) = CFMAT(3,Y)-CFMAT(4,Y)
      IF(MODE.EQ.2) X(N,N) = CFMAT(3,Y)+CFMAT(4,Y)
C
      RETURN
      END
```

```
C.....................SUBROUTINE BMSET.........................
C
C
C      PURPOSE
C
C         THE PURPOSE OF BMSET IS EXACTLY THAT OF MSET EXCEPT
C         THAT BMSET TAKES ADVANTAGE OF THE BANDED NATURE OF
C         THE FINITE DIFFERENCE MATRICES TO CONSERVE SPACE.
C
C      USAGE
C
C         CALL BMSET(X,N,MDIM,MODE,CFMAT)
C
C      DESCRIPTION OF PARAMETERS
C
C         THE PARAMETERS FOR BMSET ARE THE SAME AS THOSE FOR
C         MSET WITH THE EXCEPTION THAT THE MATRIX X MUST BE
C         DIMENSIONED (5,MDIM) IN THE CALLING PROGRAM.
C
C      NOTE:  THE PROCEDURE IS IDENTICAL TO THAT OF MSET.
C         COMMENTS HAVE THEREFORE NOT BEEN INCLUDED IN BMSET.
C
C
C.............................................................
C
      SUBROUTINE BMSET(X,N,MDIM,MODE,CFMAT)
      REAL*8 REY,Y,DELY,DFLOAT
      COMPLEX*16  CFMAT
      COMPLEX*16 A,B,G
      COMPLEX*16 X(5,MDIM)
      COMMON / COEFNT / A,B,G,REY,DELY
      DELY = 2D0/DFLOAT(N+1)
      IF(MODE.EQ.1.OR.MODE.EQ.2) DELY = 2D0/DFLOAT(2*N+1)
      IF(N.LE.MDIM) GO TO 1
      WRITE(6,9000)
 9000 FORMAT('0* * * ERROR - ARRAYS NOT DIMENSIONED LARGE',
     *       ' ENOUGH * * *')
      STOP
    1 DO 10 I=1,5
      DO 10 J=1,MDIM
   10 X(I,J) = (0D0,0D0)
      Y = 1D0-DELY
      X(3,1) = CFMAT(3,Y)+CFMAT(1,Y)
      X(4,1) = CFMAT(4,Y)
      X(5,1) = CFMAT(5,Y)
      Y = 1D0-2D0*DELY
      X(2,2) = CFMAT(2,Y)
      X(3,2) = CFMAT(3,Y)
      X(4,2) = CFMAT(4,Y)
      X(5,2) = CFMAT(5,Y)
      IL = N-2
      DO 20 I=3,IL
      Y = 1D0-DELY*DFLOAT(I)
      DO 20 J=1,5
   20 X(J,I)    = CFMAT(J,Y)
      Y = 1D0-DELY*DFLOAT(N-1)
      X(1,N-1)    = CFMAT(1,Y)
      X(2,N-1)    = CFMAT(2,Y)
      X(3,N-1)    = CFMAT(3,Y)
      X(4,N-1) = CFMAT(4,Y)
      IF(MODE.EQ.1) X(4,N-1) = CFMAT(4,Y)-CFMAT(5,Y)
      IF(MODE.EQ.2) X(4,N-1) = CFMAT(4,Y)+CFMAT(5,Y)
      Y = 1D0-DELY*DFLOAT(N)
      X(1,N)    = CFMAT(1,Y)
      X(2,N)    = CFMAT(2,Y)
      IF(MODE.EQ.1) X(2,N) = CFMAT(2,Y)-CFMAT(5,Y)
      IF(MODE.EQ.2) X(2,N) = CFMAT(2,Y)+CFMAT(5,Y)
      X(3,N) = CFMAT(3,Y)+CFMAT(5,Y)
      IF(MODE.EQ.1) X(3,N) = CFMAT(3,Y)-CFMAT(4,Y)
      IF(MODE.EQ.2) X(3,N) = CFMAT(3,Y)+CFMAT(4,Y)
      RETURN
      END
```

```
C.....................SUBROUTINE DSPLIT.......................
C
C
C      PURPOSE
C
C        DSPLIT TAKES A MATRIX OF COMPLEX*16 NUMBERS AND
C        SPLITS IT INTO TWO MATRICES, ONE CONTAINING THE REAL
C        PART OF THE ORIGINAL MATRIX, AND ONE CONTAINING THE
C        IMAGINARY PART.
C
C      USAGE
C
C        CALL DSPLIT(N,MDIM,A,AREAL,AIMAG)
C
C      DESCRIPTION OF PARAMETERS
C
C        N - THE SIZE OF THE MATRIX A, AN N BY N SQUARE
C          MATRIX.
C
C        MDIM - THE COLUMN DIMENSION OF MATRIX A
C
C        A - THE INPUT MATRIX.  MUST BE DIMENSIONED MDIM BY
C          AT LEAST N IN THE CALLING PROGRAM  (COMPLEX*16)
C
C        AREAL,AIMAG - THE OUTPUT MATRICES CONTAINING THE
C          REAL AND IMAGINARY PARTS, RESPECTIVELY, OF
C          MATRIX A.  MUST BE DIMENSIONED (MDIM,MDIM) IN THE
C          CALLING PROGRAM.
C
C      NOTES...
C
C        MATRIX A AND MATRIX AREAL MAY OVERLAP IF THEY ARE
C          DIMENSIONED IN THE CALLING PROGRAM AS FOLLOWS...
C
C               COMPLEX*16 A(MDIM,MDIM)
C               REAL*8 AREAL(MDIM,MDIM),AIMAG(MDIM,MDIM)
C               EQUIVALENCE(A(1,1),AREAL(1,1))
C
C      OTHER ROUTINES NEEDED
C
C        NONE
C
C
C
C.............................................................
C
      SUBROUTINE DSPLIT(N,MDIM,A,AR,AI)
      REAL*8 A(2,MDIM,MDIM),AR(MDIM,MDIM),AI(MDIM,MDIM)
C
      DO 1 J=1,N
      DO 1 I=1,N
      AR(I,J) = A(1,I,J)
    1 AI(I,J) = A(2,I,J)
C
      RETURN
      END
```

```
C......................SUBROUTINE DEIGEO.......................
C
C
C       PURPOSE
C
C         DEIGEO SOLVES THE LINEARIZED NAVIER-STOKES EQUATION
C         FOR POISEULLE FLOW.  THE INPUTS TO DEIGEO ARE THE
C         WAVE NOS., ALPHA AND BETA, AND THE REYNOLDS NO.
C         DEIGEO OUTPUTS THE EIGENVALUES FOR GAMMA.
C
C       USAGE
C
C         CALL DEIGEO(ALPHA,BETA,REYNO,N,MDIM,WREVEN,WIEVEN,
C           WRODD,WIODD,CDM,DM,BM,WV)
C
C       DESCRIPTION OF PARAMETERS
C
C         THE FOLLOWING MUST BE SET BY THE CALLING PROGRAM...
C           ALPHA,BETA,REYNO,N,MDIM
C
C         ALPHA - THE PERTURBATION WAVE NUMBER IN THE FLOW
C           DIRECTION (X).  (COMPLEX*16)
C
C         BETA - THE PERTURBATION WAVE NUMBER IN THE DIRECTION
C           PERPENDICULAR TO THE FLOW BUT PARALLEL TO THE
C           PLATES (Z).  (COMPLEX*16)
C
C         REYNO - THE REYNOLDS NUMBER (REAL*8)
C
C         N - THE SIZE OF THE MATRICES WHICH IS EQUAL TO
C           (ND-1)/2 WHERE ND IS THE NUMBER OF DIVISIONS
C           ACROSS THE CHANNEL.  (NOTE... DEIGEO SOLVES THE
C           PROBLEM ACROSS THE HALF CHANNEL TWICE - ONCE FOR
C           THE EIGENVALUES CORRESPONDING TO THE EVEN
C           EIGENFUNCTIONS AND ONCE FOR THOSE CORRESPONDING
C           TO THE ODD EIGENFUNCTIONS.
C
C         MDIM - THE COLUMN DIMENSION OF THE MATRICES WHICH
C           DEIGEO USES.  MDIM MUST BE .GE. N
C
C         THE FOLLOWING ARE OUTPUT BY DEIGEO
C           WREVEN,WIEVEN,WRODD,WIODD
C
C         WREVEN,WIEVEN - THE REAL AND IMAGINARY PARTS OF THE
C           EIGENVALUES CORRESPONDING TO THE EVEN
C           EIGENFUNCTIONS.  DIMENSIONED TO AT LEAST N.
C           (REAL*8)
C
C         WRODD,WIODD - THE REAL AND IMAGINARY PARTS OF THE
C           EIGENVALUES CORRESPONDING TO THE ODD
C           EIGENFUNCTIONS.  DIMENSIONED TO AT LEAST N.
C           (REAL*8)
C
C         THE FOLLOWING MATRICES MUST BE INPUT TO DEIGEO AS
C           WORKSPACE.
C
C           CDM(MDIM,MDIM)  (COMPLEX*16)
C           DM(MDIM,MDIM)  (REAL*8)
C           BM(5,MDIM)  (COMPLEX*16)
C           WV(MDIM)  (COMPLEX*16)
C
C       NOTES...
C
C         THE MATRICES CAN BE OVERLAPPED TO CONSERVE SPACE,
C         FOR EXAMPLE, FOR N = 60...
C
C                COMPLEX*16 CDM(60,30,3)
C                COMPLEX*16 DM(60,30),WV(60),BM(5,60)
C                EQUIVALENCE (DM(1,1),CDM(1,1,3)),
C                            (BM(1,1),CDM(1,1,3)),
C                            WVEC(1),CDM(1,6,3))
C
```

```
C          NOTE THAT IT IS ONLY THE ACTUAL SIZE OF THESE
C          WORKSPACES THAT IS IMPORTANT, NOT THEIR TYPE.
C
C       OTHER ROUTINES NEEDED
C
C          THE FOLLOWING ARE CALLED BY DEIGEO
C
C             CHM1E1,CHM2E2,MSET,CDMTIN,BMSET,MULDBM,DSPLIT,
C             EHESSC,ELRH1C
C
C......................................................................
C
      SUBROUTINE DEIGEO(ALPHA,BETA,REYNO,N,MDIM,
     *     WREVEN,WIEVEN,WRODD,WIODD,CDM,DM,BM,WV)
      IMPLICIT COMPLEX*16(A-H,O-Z)
      DIMENSION IVEC(100)
      REAL*8 WREVEN(1),WIEVEN(1),WRODD(1),WIODD(1)
      REAL*8 CDM(1),DM(1),BM(1),WV(1)
      REAL*8 REY,DELY,REYNO
      COMMON / COEFNT / A,B,G,REY,DELY
      EXTERNAL CHM1E1,CHM2E1
C
C THIS SUBROUTINE SOLVES THE EQUATION  YV = GXV  WHERE
C X AND Y ARE MATRICES, V IS THE EIGENVECTOR AND G IS THE
C EIGENVALUE.  THE EIGENVALUES ARE DETERMINED AND PASSED
C BACK TO THE CALLING PROGRAM IN WRODD, WIODD, WREVEN AND
C WIEVEN.
C
      A = ALPHA
      B = BETA
      REY = REYNO
C
C SET UP MATRIX X FOR ODD EIGENVECTORS.
C
      CALL MSET(CDM,N,MDIM,1,CHM2E1)
C
C INVERT MATRIX X.
C
      CALL CDMTIN(N,CDM,MDIM,DETERM)
C
C SET UP MATRIX Y IN BAND STORAGE MODE FOR ODD EIGENVECTORS.
C
      CALL BMSET(BM,N,MDIM,1,CHM1E1)
C
C MULTIPLY MATRIX Y BY THE INVERSE OF MATRIX X TO CONVERT
C TO THE STANDARD EIGENVALUE PROBLEM WHICH HAS THE FORM
C (Z-G)V = 0   WHERE  Z = (Y)(INVERSE(X)).
C
      CALL MULDBM(CDM,BM,N,5,MDIM,WV)
C
C SPLIT MATRIX INTO REAL AND IMAGINARY PARTS AND CALL
C THE SUBROUTINES TO FIND THE EIGENVALUES.
C
      CALL DSPLIT(N,MDIM,CDM,CDM,DM)
      CALL EHESSC(CDM,DM,1,N,N,MDIM,IVEC)
      CALL ELRH1C(CDM,DM,1,N,N,MDIM,WRODD,WIODD,INERR,IER)
      IF(INERR.NE.0) WRITE(6,9000) INERR,IER
 9000 FORMAT('0ERROR NUMBER',I7,'  ON EIGENVALUE',I7,///)
C
C REPEAT THE SOLUTION FOR EIGENVALUES FOR THE EVEN
C EIGENVECTORS
C
      CALL MSET(CDM,N,MDIM,2,CHM2E1)
      CALL CDMTIN(N,CDM,MDIM,DETERM)
      CALL BMSET(BM,N,MDIM,2,CHM1E1)
      CALL MULDBM(CDM,BM,N,5,MDIM,WV)
      CALL DSPLIT(N,MDIM,CDM,CDM,DM)
      CALL EHESSC(CDM,DM,1,N,N,MDIM,IVEC)
      CALL ELRH1C(CDM,DM,1,N,N,MDIM,WREVEN,WIEVEN,INERR,IER)
      IF(INERR.NE.0) WRITE(6,9000) INERR,IER
      RETURN
      END
```

```
C.......................................................................
C
C     SUBROUTINE CDMTIN (CATEGORY F-1)
C
C     PURPOSE
C
C        INVERT A COMPLEX*16 MATRIX
C
C     USAGE
C
C        CALL CDMTIN(N,A,NDIM,DETERM)
C
C     DESCRIPTION OF PARAMETERS
C
C        N      - ORDER OF COMPLEX*16 MATRIX TO BE INVERTED
C                 (INTEGER) MAXIMUM 'N' IS 100
C
C        A      - COMPLEX*16 INPUT MATRIX (DESTROYED).  THE
C                 INVERSE OF 'A' IS RETURNED IN ITS PLACE
C
C        NDIM   - THE SIZE TO WHICH 'A' IS DIMENSIONED
C                 (ROW DIMENSION OF 'A' ACTUALLY APPEARING
C                 IN THE DIMENSION STATMENT OF USER'S
C                 CALLING PROGRAM)
C
C        DETERM - COMPLEX*16 VALUE OF DETERMINANT OF 'A'
C                 RETURNED BY CDMTIN.
C
C     REMARKS
C
C        MATRIX 'A' MUST BE A COMPLEX*8 GENERAL MATRIX
C        IF MATRIX 'A' IS SINGULAR THAT MESSAGE IS PRINTED
C        'N' MUST BE .LE. NDIM
C
C     SUBROUTINES AND FUNCTIONS REQUIRED
C
C        ONLY BUILT-IN FORTRAN FUNCTIONS
C
C     METHOD
C
C        GAUSSIAN ELIMINATION WITH COLUMN PIVOTING IS USED.
C        THE DETERMINANT IS ALSO CALCULATED. A DETERMINANT
C                 OF ZERO INDICATES THAT MATRIX 'A' IS
C                 SINGULAR.
C
C.......................................................................
C
      SUBROUTINE CDMTIN (N,A,NDIM,DETERM)
      IMPLICIT REAL*8 (A-H,O-Z)
      COMPLEX*16 A(NDIM,NDIM),PIVOT(100),AMAX,T,SWAP,
     *    DETERM,U
      INTEGER*4 IPIVOT(100),INDEX(100,2)
      REAL*8 TEMP,ALPHA(100)
C
C      INITIALIZATION
C
      DETERM = (1D0,0D0)
      DO 20 J=1,N
      ALPHA(J) = 0D0
      DO 10 I=1,N
   10 ALPHA(J)=ALPHA(J)+A(J,I)*DCONJG(A(J,I))
      ALPHA(J)=DSQRT(ALPHA(J))
   20 IPIVOT(J)=0
      DO 600 I=1,N
C
C      SEARCH FOR PIVOT ELEMENT
C
      AMAX = (0D0,0D0)
      DO 105 J=1,N
      IF (IPIVOT(J)-1) 60,105,60
   60 DO 100 K=1,N
      IF (IPIVOT(K)-1) 80,100,740
```

```
   8C TEMP=AMAX*DCONJG(AMAX)-A(J,K)*DCONJG(A(J,K))
      IF(TEMP) 85, 85, 100
   85 IROW=J
      ICOLUM=K
      AMAX=A(J,K)
  100 CONTINUE
  105 CONTINUE
      IPIVOT(ICOLUM)=IPIVOT(ICOLUM)+1
C
C      INTERCHANGE ROWS TO PUT PIVOT ELEMENT ON DIAGONAL
C
      IF(IROW-ICOLUM) 140, 260, 140
  140 DETERM=DETERM
      DO 200 L=1,N
      SWAP=A(IROW,L)
      A(IROW,L)=A(ICOLUM,L)
  200 A(ICOLUM,L)=SWAP
      SWAP=ALPHA(IROW)
      ALPHA(IROW)=ALPHA(ICOLUM)
      ALPHA(ICOLUM)=SWAP
  260 INDEX(I,1)=IROW
      INDEX(I,2)=ICOLUM
      PIVOT(I)=A(ICOLUM,ICOLUM)
      U=PIVOT(I)
      TEMP=PIVOT(I)*DCONJG(PIVOT(I))
      IF(TEMP) 330, 720, 330
C
C      DIVIDE PIVOT ROW BY PIVOT ELEMENT
C
  330 A(ICOLUM,ICOLUM) = (1D0,0D0)
      DO 350 L=1,N
      U=PIVOT(I)
  350 A(ICOLUM,L)=A(ICOLUM,L)/U
C
C      REDUCE NON-PIVOT ROWS
C
  380 DO 550 L1=1,N
      IF(L1-ICOLUM) 400, 550, 400
  400 T=A(L1,ICOLUM)
      A(L1,ICOLUM) = (0D0,0D0)
      DO 450 L=1,N
      U=A(ICOLUM,L)
  450 A(L1,L)=A(L1,L)-U*T
  550 CONTINUE
  600 CONTINUE
C
C      INTERCHANGE COLUMNS
C
  620 DO 710 I=1,N
      L=N+1-I
      IF(INDEX(L,1)-INDEX(L,2)) 630, 710, 630
  630 JROW=INDEX(L,1)
      JCOLUM=INDEX(L,2)
      DO 705 K=1,N
      SWAP=A(K,JROW)
      A(K,JROW)=A(K,JCOLUM)
      A(K,JCOLUM)=SWAP
  705 CONTINUE
  710 CONTINUE
      RETURN
  720 WRITE(6,730)
  730 FORMAT(20H  MATRIX IS SINGULAR)
  740 RETURN
      END
```

```
C.......................SUBROUTINE DFIT2.......................
C
C
C     PURPOSE
C
C       THIS SUBROUTINE TAKES 3 POINTS (X1,Y), (X2,Y2),
C       AND (X3,Y3), AND USES A PARABOLIC FIT TO FIND THE
C       EXTREMAL VALUE OF Y AND THE CORRESPONDING VALUE FOR
C       X WHICH ARE RETURNED IN XM AND YM.  IT ALSO TAKES AN
C       INPUT VALUE, YVAL, AND COMPUTES THE TWO SOLUTIONS
C       FOR X TO THE PARABOLIC INTERPOLATION, TAKES THE
C       SOLUTION NEAREST TO X1, AND RETURNS IT IN XNEXT.
C       IF THERE IS NO REAL SOLUTION TO THE EQUATION, THEN
C       XNEXT IS SET TO 0.0.
C
C     USAGE
C
C       CALL DFIT2(X1,X2,X3,Y1,Y2,Y3,YVAL,XNEXT,XM,YM)
C
C     DESCRIPTION OF PARAMETERS
C
C       THE FOLLOWING MUST BE SET BY THE CALLING PROGRAM
C         X1,X2,X3,Y1,Y2,Y3,YVAL
C
C       X1,X2,X3, - THE X VALUES OF 3 POINTS (REAL*8)
C       Y1,Y2,Y3 - THE Y VALUES OF 3 POINTS (REAL*8)
C       YVAL - A VALUE OF Y FOR WHICH THE CORRESPONDING
C         X VALUE IS DESIRED (REAL*8)
C
C       THE FOLLOWING ARE SET BY DFIT2
C         XNEXT,XM,YM
C
C       XNEXT - SET TO THE VALUE OF X NEAREST X1
C         CORRESPONDING TO YVAL (REAL*8)
C       XM,YM - SET TO THE MIN Y VALUE FROM THE SECOND
C         ORDER INTERPOLATION, WITH THE CORRESPONDING X
C         VALUE (REAL*8)
C
C
C...............................................................
C
      SUBROUTINE DFIT2(X1,X2,X3,Y1,Y2,Y3,YVAL,XNEXT,XM,YM)
      IMPLICIT REAL*8(A-Z)
C
      A = ((Y1-Y2)/(X1-X2)-(Y1-Y3)/(X1-X3))
     *    /((X1**2-X2**2)/(X1-X2)-(X1**2-X3**2)/(X1-X3))
      B = (Y1-Y2-A*(X1**2-X2**2))/(X1-X2)
      C = Y1-A*X1**2-B*X1
      XM = -B/(2D0*A)
      YM = A*XM**2+B*XM+C
      D = B**2-4D0*A*(C-YVAL)
      IF(D.GE.0D0) GO TO 10
      XNEXT = 0D0
      RETURN
C
   10 XN1 = (-B+DSQRT(D))/(2D0*A)
      XN2 = (-B-DSQRT(D))/(2D0*A)
      XNEXT = XN1
      IF(DABS(XN2-X1).LT.DABS(XN1-X1)) XNEXT = XN2
      RETURN
      END
```

```
C.............................................................
C PRCGRAM #R1A
C
C                 PROGRAM TO PRINT EIGENVALUES
C              FOR THE 3-D CYLINDRICAL FLOW PROBLEM
C                        N = 0       FOR G
C
C.............................................................
C
C        INPUT VALUES ARE N, REY, AR, AND AI.
C          (USING NAMELIST 'LIST')
C
C.............................................................
      IMPLICIT REAL*8(A-H,O-Z)
      CCMPLEX*16 A,B,DETERM,G
      CCMPLEX*16 XMAT(31,31),YMAT(31,31),WV(31)
      CCMPLEX*16 CGM1E2,CGM2E2
      REAL*8 GR(31),GI(31)
      REAL*4 GR4(31),GI4(31)
      DIMENSICN IVEC(100)
      CCMMCN / CCEFNT / A,B,G,REY,DELR
      NAMELIST / LIST / N,REY,AR,AI
      EXTERNAL CGM1E2,CGM2E2
C
      MDIM = 31
      N = 30
      REY = 6000D0
      AR = 0D0
      AI = 1D0
      B = (0D0,0D0)
C
    1 READ(5,LIST,END=100)
      A = DCMPLX(AR,AI)
      WRITE(6,9000)
 9000 FCRMAT('1')
      WRITE(6,9001) N,REY,A
 9001 FCRMAT(' CCMPONENT G IN EQUATICN 2',/,'ON =',I4,/,
     *     ' REY=',F10.2,8X,'ALPHA =',2F12.7,8X,'NBI = 0')
C
      NP1 = N+1
      CALL SET2(XMAT,NP1,1,MDIM,CGM2E2,1,1)
      CALL CCMTIN(N,XMAT,MDIM,DETERM)
      CALL SET2(YMAT,NP1,1,MDIM,CGM1E2,1,1)
      CALL MULM(XMAT,YMAT,N,MDIM,WV)
      CALL DSPLIT(N,MDIM,XMAT,XMAT,YMAT)
      CALL EHESSC(XMAT,YMAT,1,N,N,MDIM,IVEC)
      CALL ELRH1C(XMAT,YMAT,1,N,N,MDIM,GR,GI,INERR,IER)
      IF(INERR.NE.0) WRITE(6,9010) INERR,IER
 9010 FORMAT('0* * * ERROR NUMBER',I7,' ON EIGENVALUE',
     *     I7,' * * *',///)
C
      WRITE(6,9002)
 9002 FCRMAT(///,16X,'GAMMA REAL',5X,'GAMMA IMAG')
      DC 10 I=1,N
      GR4(I) = SNGL(GR(I))
      GI4(I) = SNGL(GI(I))
   10 WRITE(6,9003) I,GR(I),GI(I)
 9003 FCRMAT('0',I10,1P2D15.4)
C
      WRITE(6,9000)
      CALL FLCTP(GR4,GI4,N,0)
      WRITE(6,9001) N,REY,A
C
      GO TO 1
  100 WRITE(6,9000)
      STOP
      END
```

```
C.......................................................................
C  PROGRAM #R1B
C
C                    PROGRAM TO PRINT EIGENVALUES
C               FOR THE 3-D CYLINDRICAL FLOW PROBLEM
C                          N = 0       FOR H
C
C.......................................................................
C
C          INPUT VALUES ARE N, REY, AR, AND AI.
C            (USING NAMELIST 'LIST')
C
C.......................................................................
      IMPLICIT REAL*8(A-H,O-Z)
      COMPLEX*16 A,B,DETERM,G
      COMPLEX*16 XMAT(31,31),YMAT(31,31),WV(31)
      COMPLEX*16 CHM1E1,CHM2E1
      REAL*8 GR(31),GI(31)
      REAL*4 GR4(31),GI4(31)
      DIMENSION IVEC(100)
      COMMON / COEFNT / A,B,G,REY,DELR
      NAMELIST / LIST / N,REY,AR,AI
      EXTERNAL CHM1E1,CHM2E1
C
      MDIM = 31
      N = 30
      REY = 6000D0
      AR = 0D0
      AI = 1D0
      B = (0D0,0D0)
C
    1 READ(5,LIST,END=100)
      A = DCMPLX(AR,AI)
      WRITE(6,9000)
 9000 FORMAT('1')
      WRITE(6,9001) N,REY,A
 9001 FORMAT(' COMPONENT H IN EQUATION 1',/,'0N =',I4,/,
     *       ' REY=',F10.2,8X,'ALPHA =',2F12.7,8X,'NBI = 0')
C
      NP1 = N+1
      CALL SET2(XMAT,NP1,1,MDIM,CHM2E1,1,1)
      CALL CCMTIN(N,XMAT,MDIM,DETERM)
      CALL SET2(YMAT,NP1,1,MDIM,CHM1E1,1,1)
      YMAT(N,N) = YMAT(N,N) + CHM1E1(5,DELR)
      CALL MULM(XMAT,YMAT,N,MDIM,WV)
      CALL DSPLIT(N,MDIM,XMAT,XMAT,YMAT)
      CALL EHESSC(XMAT,YMAT,1,N,N,MDIM,IVEC)
      CALL ELRH1C(XMAT,YMAT,1,N,N,MDIM,GR,GI,INERR,IER)
      IF(INERR.NE.0) WRITE(6,9010) INERR,IER
 9010 FORMAT('0* * * ERROR NUMBER',I7,' ON EIGENVALUE',
     *       I7,' * * *',///)
C
      WRITE(6,9002)
 9002 FORMAT(///,16X,'GAMMA REAL',5X,'GAMMA IMAG')
      DO 10 I=1,N
      GR4(I) = SNGL(GR(I))
      GI4(I) = SNGL(GI(I))
   10 WRITE(6,9003) I,GR(I),GI(I)
 9003 FORMAT('0',I10,1P2D15.4)
C
      WRITE(6,9000)
      CALL FLOTP(GR4,GI4,N,0)
      WRITE(6,9001) N,REY,A
C
      GO TO 1
  100 WRITE(6,9000)
      STOP
      END
```

```
C.......................................................................
C PROGRAM #R2
C
C                 PROGRAM TO PRINT EIGENVALUES
C             FOR THE 3-D CYLINDRICAL FLOW PROBLEM
C                            N = 1
C
C
C.......................................................................
      IMPLICIT REAL*8(A-H,O-Z)
      COMPLEX*16 A,B,DETERM,G
      COMPLEX*16 XMAT(60,60),YMAT(60,60),WV(60)
      COMPLEX*16 CFM1E1,CFM2E1,CFM1E2,CFM2E2,CGM1E1,CGM2E1,
     *    CGM1E2,CGM2E2,CHM1E1,CHM2E1,CHM1E2,CHM2E2
      REAL*8 GR(60),GI(60)
      REAL*4 GR4(60),GI4(60)
      DIMENSION IVEC(100)
      COMMON / COEFNT / A,B,G,REY,DELR
      NAMELIST / LIST / N,REY,AR,AI
      EXTERNAL    CFM1E1,CFM2E1,CFM1E2,CFM2E2,CGM1E1,CGM2E1,
     *    CGM1E2,CGM2E2,CHM1E1,CHM2E1,CHM1E2,CHM2E2
C
      MDIM = 60
      N = 30
      REY = 6000D0
      AR = 0D0
      AI = 1D0
      B = (0D0,1D0)
C
    1 READ(5,LIST,END=100)
      A = DCMPLX(AR,AI)
      WRITE(6,9000)
 9000 FORMAT('1')
      WRITE(6,9001) N,REY,A
 9001 FORMAT(' N =',I4,/,' REY =',F10.2,8X,'ALPHA =',2F12.7,
     *    8X,'NBI = 1',/,' FOR VELOCITY VECTOR POTENTIAL ',
     *       'H SET EQUAL TO ZERO')
C
      MSIZE = 2*(N-1)
      CALL SET2(XMAT,N,1,MDIM,CFM2E1,1,1)
      CALL SET2(XMAT,N,1,MDIM,CGM2E1,1,N)
      CALL SET2(XMAT,N,1,MDIM,CFM2E2,N,1)
      CALL SET2(XMAT,N,1,MDIM,CGM2E2,N,N)
C
      XMAT(N-1,N-1) = XMAT(N-1,N-1) + CFM2E1(5,DELR)
      XMAT(MSIZE,N-1) = XMAT(MSIZE,N-1) + CFM2E2(5,DELR)
C
      CALL CDMTIN(MSIZE,XMAT,MDIM,DETERM)
C
      CALL SET2(YMAT,N,1,MDIM,CFM1E1,1,1)
      CALL SET2(YMAT,N,1,MDIM,CGM1E1,1,N)
      CALL SET2(YMAT,N,1,MDIM,CFM1E2,N,1)
      CALL SET2(YMAT,N,1,MDIM,CGM1E2,N,N)
C
      YMAT(N-1,N-1) = YMAT(N-1,N-1) + CFM1E1(5,DELR)
      YMAT(MSIZE,N-1) = YMAT(MSIZE,N-1) + CFM1E2(5,DELR)
C
      CALL MULM(XMAT,YMAT,MSIZE,MDIM,WV)
      CALL DSPLIT(MSIZE,MDIM,XMAT,XMAT,YMAT)
      CALL EHESSC(XMAT,YMAT,1,MSIZE,MSIZE,MDIM,IVEC)
      CALL ELRHIC(XMAT,YMAT,1,MSIZE,MSIZE,MDIM,
     *    GR,GI,INERR,IER)
      IF(INERR.NE.0) WRITE(6,9010) INERR,IER
 9010 FORMAT('0* * * ERROR NUMBER',I7,' ON EIGENVALUE',
     *    I7,' * * *',///)
C
      WRITE(6,9002)
 9002 FORMAT(///,16X,'GAMMA REAL',10X,'GAMMA IMAG')
      DO 10 I=1,MSIZE
      GR4(I) = SNGL(GR(I))
      GI4(I) = SNGL(GI(I))
   10 WRITE(6,9003) I,GR(I),GI(I)
```

```
 9003 FORMAT('0',I10,1P2D20.10)
C
      WRITE(6,9000)
      CALL PLOTP(GR4,GI4,MSIZE,0)
      WRITE(6,9001) N,REY,A
C
      GO TO 1
  100 WRITE(6,9000)
      STOP
      END
```

```
      FUNCTION CHM1E1(K,R)
      IMPLICIT COMPLEX*16 (A-H,O-Z)
      COMMON / COEFNT / A,B,G,REY,DEL
      REAL*8 REY,R,DEL
C
      T1(R)  =  A**2+(B/R)**2
      T2(R)  =  T1(R)/REY-A*2D0*(1D0-R**2)
C
      CF4M1(R)  =  B/(R*REY)
      CF3M1(R)  =  2D0*B/(R**2*REY)
      CF2M1(R)  =  B*T2(R)/R-B/(R**3*REY)+B*T1(R)/(R*REY)
      CF1M1(R)  =  B*T2(R)/R**2+B/(R**4*REY)-3D0*B**3/(R**4
     *     *REY) +A**2*B/(R**2*REY)
      CF0M1(R)  =  T2(R)*B*T1(R)/R +4D0*B**3/(R**5*REY)
C
      CF2M2(R)  =  B/R
      CF1M2(R)  =  B/R**2
      CF0M2(R)  =  B*T1(R)/R
C
      CG2M1(R)  =  -4D0*A*B/(R**2*REY)
      CG1M1(R)  =  4D0*A*B/(R**3*REY)
      CG0M1(R)  =  -4D0*A*B/(R**4*REY)-2D0*A*B*(T2(R)
     *     +T1(R)/REY)/R**2
C
      CG0M2(R)  =  -2D0*A*B/R**2
C
      CH4M1(R)  =  -A/REY
      CH3M1(R)  =  -2D0*A/(R*REY)
      CH2M1(R)  =  -A*T1(R)/REY-A*T2(R)+3D0*A/(R**2*REY)
      CH1M1(R)  =  3D0*A*B**2/(R**3*REY)-A*T2(R)/R-3D0*A
     *     /(R**3*REY)-A**3/(R*REY)
      CH0M1(R)  =  T2(R)*(A/R**2-A*T1(R))+A*B**2/(R**4*REY)
     *     +3D0*A/(R**4*REY)+A**3/(R**2*REY)
C
      CH2M2(R)  =  -A
      CH1M2(R)  =  -A/R
      CH0M2(R)  =  A/R**2-A*T1(R)
C
      GO TO (11,12,13,14,15),K
   11 CHM1E1  =  CH4M1(R)/DEL**4-CH3M1(R)/(2D0*DEL**3)
      GO TO 100
   12 CHM1E1  =  -4D0*CH4M1(R)/DEL**4+2D0*CH3M1(R)/(2D0*DEL**3
     *    )+CH2M1(R)/DEL**2-CH1M1(R)/(2D0*DEL)
      GO TO 100
   13 CHM1E1  =  6D0*CH4M1(R)/DEL**4-2D0*CH2M1(R)/DEL**2
     *     +CH0M1(R)
      GO TO 100
   14 CHM1E1  =  -4D0*CH4M1(R)/DEL**4-2D0*CH3M1(R)/(2D0*DEL**3
     *    )+CH2M1(R)/DEL**2+CH1M1(R)/(2D0*DEL)
      GO TO 100
   15 CHM1E1  =  CH4M1(R)/DEL**4+CH3M1(R)/(2D0*DEL**3)
  100 RETURN
C
      ENTRY CHM2E1(K,R)
      GO TO (21,22,23,24,21),K
   21 CHM2E1  =  (0D0,0D0)
      GO TO 200
   22 CHM2E1  =  CH2M2(R)/DEL**2-CH1M2(R)/(2D0*DEL)
      GO TO 200
   23 CHM2E1  =  -2D0*CH2M2(R)/DEL**2+CH0M2(R)
      GO TO 200
   24 CHM2E1  =  CH2M2(R)/DEL**2+CH1M2(R)/(2D0*DEL)
  200 RETURN
C
      ENTRY CGM1E1(K,R)
      GO TO (31,32,33,34,31),K
   31 CGM1E1  =  (0D0,0D0)
      GO TO 300
   32 CGM1E1  =  CG2M1(R)/DEL**2-CG1M1(R)/(2D0*DEL)
      GO TO 300
   33 CGM1E1  =  -2D0*CG2M1(R)/DEL**2+CG0M1(R)
      GO TO 300
```

```
   34 CGM1E1 = CG2M1(R)/DEL**2+CG1M1(R)/(2D0*DEL)
  300 RETURN
C
      ENTRY CGM2E1(K,R)
      GO TO (41,41,42,41,41),K
   41 CGM2E1 = (0D0,0D0)
      GO TO 400
   42 CGM2E1 = CG0M2(R)
  400 RETURN
C
      ENTRY CFM1E1(K,R)
      GO TO (51,52,53,54,55),K
   51 CFM1E1 = CF4M1(R)/DEL**4-CF3M1(R)/(2D0*DEL**3)
      GO TO 500
   52 CFM1E1 = -4D0*CF4M1(R)/DEL**4+2D0*CF3M1(R)/(2D0*DEL**3
     *    )+CF2M1(R)/DEL**2-CF1M1(R)/(2D0*DEL)
      GO TO 500
   53 CFM1E1 = 6D0*CF4M1(R)/DEL**4-2D0*CF2M1(R)/DEL**2
     *      +CF0M1(R)
      GO TO 500
   54 CFM1E1 = -4D0*CF4M1(R)/DEL**4-2D0*CF3M1(R)/(2D0*DEL**3
     *    )+CF2M1(R)/DEL**2+CF1M1(R)/(2D0*DEL)
      GO TO 500
   55 CFM1E1 = CF4M1(R)/DEL**4+CF3M1(R)/(2D0*DEL**3)
  500 RETURN
C
      ENTRY CFM2E1(K,R)
      GO TO (61,62,63,64,61),K
   61 CFM2E1 = (0D0,0D0)
      GO TO 600
   62 CFM2E1 = CF2M2(R)/DEL**2-CF1M2(R)/(2D0*DEL)
      GO TO 600
   63 CFM2E1 = -2D0*CF2M2(R)/DEL**2+CF0M2(R)
      GO TO 600
   64 CFM2E1 = CF2M2(R)/DEL**2+CF1M2(R)/(2D0*DEL)
  600 RETURN
      END
```

```
      FUNCTICN CHM1E2(K,R)
      IMPLICIT CCMPLEX*16(A-H,O-Z)
      CCMMCN / CCEFNT / A,B,G,REY,DEL
      REAL*8 REY,R,DEL
C
      T1(R) = A**2+(B/R)**2
      T2(R) = T1(R)/REY-A*2D0*(1D0-R**2)
C
      CF3M1(R) = A/REY
      CF2M1(R) = A/(R*REY)
      CF1M1(R) = A*(T2(R)-1D0/(R**2*REY))
      CF0M1(R) = 4D0*B**2/R-2D0*A*B**2/(R**3*REY)
C
      CF1M2(R) = A
C
      CG2M1(R) = -T1(R)/REY
      CG1M1(R) = 2D0*B**2/(R**3*REY)-T1(R)/(R*REY)
      CG0M1(R) = T1(R)*(1D0/(R**2*REY)-T2(R))-2D0*B**2
     *      /(R**4*REY)
C
      CG0M2(R) = -T1(R)
C
      CH3M1(R) = B/(R*REY)
      CH2M1(R) = 2D0*B/(R**2*REY)
      CH1M1(R) = B*(T2(R)-1D0/(R**2*REY))/R
      CH0M1(R) = -4D0*A*B+B*T2(R)/R**2+B/(R**4*REY)
     *      +2D0*A**2*B/(R**2*REY)
C
      CH1M2(R) = B/R
      CH0M2(R) = B/R**2
C
      GC TC (11,12,13,14,15),K
   11 CHM1E2 = -CH3M1(R)/(2D0*DEL**3)
      GC TO 100
   12 CHM1E2 = 2D0*CH3M1(R)/(2D0*DEL**3)+CH2M1(R)/DEL**2
     *      -CH1M1(R)/(2D0*DEL)
      GO TO 100
   13 CHM1E2 = -2D0*CH2M1(R)/DEL**2+CH0M1(R)
      GO TO 100
   14 CHM1E2 = -2D0*CH3M1(R)/(2D0*DEL**3)+CH2M1(R)/DEL**2
     *      +CH1M1(R)/(2D0*DEL)
      GO TO 100
   15 CHM1E2 = CH3M1(R)/(2D0*DEL**3)
  100 RETURN
C
      ENTRY CHM2E2(K,R)
      GC TO (21,22,23,24,21),K
   21 CHM2E2 = (0D0,0D0)
      GC TC 200
   22 CHM2E2 = -CH1M2(R)/(2D0*DEL)
      GO TO 200
   23 CHM2E2 = CH0M2(R)
      GC TO 200
   24 CHM2E2 = CH1M2(R)/(2D0*DEL)
  200 RETURN
C
      ENTRY CGM1E2(K,R)
      GC TO (31,32,33,34,31),K
   31 CGM1E2 = (0D0,0D0)
      GC TC 300
   32 CGM1E2 = CG2M1(R)/DEL**2-CG1M1(R)/(2D0*DEL)
      GO TO 300
   33 CGM1E2 = -2D0*CG2M1(R)/DEL**2+CG0M1(R)
      GC TO 300
   34 CGM1E2 = CG2M1(R)/DEL**2+CG1M1(R)/(2D0*DEL)
  300 RETURN
C
      ENTRY CGM2E2(K,R)
      GC TO (41,41,42,41,41),K
   41 CGM2E2 = (0D0,0D0)
      GO TO 400
   42 CGM2E2 = CG0M2(R)
```

```
  40C RETURN
C
      ENTRY CFM1E2(K,R)
      GC TO (51,52,53,54,55),K
   51 CFM1E2 = -CF3M1(R)/(2D0*DEL**3)
      GC TO 500
   52 CFM1E2 = 2D0*CF3M1(R)/(2D0*DEL**3)+CF2M1(R)/DEL**2
     *     -CF1M1(R)/(2D0*DEL)
      GO TO 5C0
   53 CFM1E2 = -2D0*CF2M1(R)/DEL**2+CF0M1(R)
      GC TO 500
   54 CFM1E2 = -2D0*CF3M1(R)/(2D0*DEL**3)+CF2M1(R)/DEL**2
     *     +CF1M1(R)/(2D0*DEL)
      GC TO 5CC
   55 CFM1E2 = CF3M1(R)/(2D0*DEL**3)
  50C RETURN
C
      ENTRY CFM2E2(K,R)
      GC TC (61,62,61,64,61),K
   61 CFM2E2 = (0D0,0D0)
      GC TO 600
   62 CFM2E2 = -CF1M2(R)/(2D0*DEL)
      GC TO 600
   64 CFM2E2 = CF1M2(R)/(2D0*DEL)
  600 RETURN
      END
```

```
      SUBROUTINE SET2(X,N,MINR,MDIM,CFMAT,NQV,NQH)
      REAL*8 REY,R,DEL,DFLOAT
      COMPLEX*16 A,B,G
      COMMON / COEFNT / A,B,G,REY,DEL
      COMPLEX*16 CFMAT
      COMPLEX*16 X(MDIM,MDIM)
      DEL = 1D0/DFLOAT(N)
C
      MV = NQV-1
      MH = NQH-1
      IF(N+MV.LE.MDIM.AND.N+MH.LE.MDIM) GO TO 1
      WRITE(6,9000)
 9000 FORMAT('0* * * ERROR - ARRAYS NOT DIMENSIONED LARGE',
     *     ' ENOUGH * * *')
      STOP
    1 DO 10 I=1,N
      DO 10 J=1,N
   10 X(I+MV,J+MH) = (0D0,0D0)
      R = 1D0-DEL
      X(1+MV,1+MH) = CFMAT(3,R)+CFMAT(1,R)
      X(1+MV,2+MH) = CFMAT(4,R)
      X(1+MV,3+MH) = CFMAT(5,R)
      R = 1D0-2D0*DEL
      X(2+MV,1+MH) = CFMAT(2,R)
      X(2+MV,2+MH) = CFMAT(3,R)
      X(2+MV,3+MH) = CFMAT(4,R)
      X(2+MV,4+MH) = CFMAT(5,R)
      IL = N-2
      DO 20 I=3,IL
      K = I-3
      R = 1D0-DEL*DFLOAT(I)
      DO 20 J=1,5
   20 X(I+MV,K+J+MH) = CFMAT(J,R)
      R = DEL
      X(N-1+MV,N-3+MH) = CFMAT(1,R)
      X(N-1+MV,N-2+MH) = CFMAT(2,R)
      X(N-1+MV,N-1+MH) = CFMAT(3,R)
      X(N-1+MV,N+MH) = CFMAT(4,R)
      IF(MINR.NE.0) GO TO 30
      R = 0D0
      X(N+MV,N-2+MH) = CFMAT(1,R)
      X(N+MV,N-1+MH) = CFMAT(2,R)
      X(N+MV,N+MH) = CFMAT(3,R)
   30 RETURN
      END
```

BIBLIOGRAPHY

Betchov, R. and Criminale, W. O. Jr., Stability of Parallel Flows, Academic Press, 1967

Davey, A., Hocking, L. M., and Stewartson, K., "On the nonlinear evolution of three-dimensional disturbances in plane Poiseuille flow", Journal of Fluid Mechanics, v. 63, part 3, p. 529, 12 November 1973

Davey, A., and Drazin, P. G., "The stability of Poiseuille flow in a pipe", Journal of Fluid Mechanics, v. 36, part 2, p. 209, 22 August 1968

Dowell, E. H., "Non-linear theory of unstable plane Poiseuille flow", Journal of Fluid Mechanics, v. 38, part 2, p. 401, 6 January 1969

Garg, V. K., and Rouleau, W. T., "Linear spatial stability of pipe Poiseuille flow", Journal of Fluid Mechanics, v. 54, part 1, p. 113, 25 November 1971

Gaster, M., "On the generation of spatially growing waves in a boundary layer", Journal of Fluid Mechanics, v. 22, part 3, p. 433, 10 June 1964

George, W. D., and Hellums, J. D., "Hydronamic stability in plane Poiseuille flow with finite amplitude disturbances", Journal of Fluid Mechanics, v. 51, part 4, p. 687, 6 July 1971

George, W. D., Hellums, J. D., and Martin, B., "Finite-amplitude neutral disturbances in plane Poiseuille flow", Journal of Fluid Mechanics, v. 63, part 2, p. 765, 3 December 1973

Gill, A. E., "The least-damped disturbance to Poiseuille flow in a circular pipe", Journal of Fluid Mechanics, v. 61, part 1, p. 97, 27 March 1973

Graebel, W. P., "The stability of pipe flow. Part 1. Asymptotic analysis for small wave numbers", Journal of Fluid Mechanics, v. 43, part 2, p. 279, 26 December 1969

Grosch, C. E. and Salwen, H., "The stability of steady and time-dependent plane Poiseuille flow", Journal of Fluid Mechanics, v. 34, part 1, p. 177, 22 April 1968

Huang, L. M, and Chen, T. S., "Stability of developing flow subject to non-axisymmetric disturbances", Journal of Fluid Mechanics, v. 63, part 1, p. 183, 16 April 1973

Kao, T. W. and Park, C., "Experimental investigations of the stability of channel flows. Part 1. Flow of a single liquid in a rectangular channel", Journal of Fluid Mechanics, v. 43, part 1, p. 145, 22 September 1969

Leite, R. J., An Experimental Investigation of the Stability of Axially Symmetric Poiseuille Flow, Report No. OSR-TR-56-2, Air Force Contract AF18(600)-350, November 1956

McIntire, L. V., and Lin, C. H., "Finite amplitude instability of second-order fluids in plane Poiseuille flow", Journal of Fluid Mechanics, v. 52, part 2, p. 273, 31 March 1971

Orszag, S. A., "Accurate solution of the Orr-Sommerfeld stability equation", Journal of Fluid Mechanics, v. 50,

part 4, p. 689, 27 May 1971

Pekeris, C. L., and Shkoller, B., "Stability of plane Poiseuille flow to periodic disturbances of finite amplitude", Journal of Fluid Mechanics, v. 39, part 3, p. 611, 5 May 1969

Pekeris, C. L. and Shkoller, B., "The neutral curves for periodic perturbations of finite amplitude of plane Poiseuille flow", Journal of Fluid Mechanics, v. 39, part 3, p. 629, 22 May 1969

Salwen, H., and Grosch, C. E., "The stability of Poiseuille flow in a pipe of circular cross-section", Journal of Fluid Mechanics, v. 54, part 1, p. 93, 6 March 1972

Schlichting, H., Boundary Layer Theory, McGraw-Hill, 1968, page 516

Squire, H. B., "On the stability of three-dimensional disturbances of viscous flow between walls", Proc. Roy. Soc., v. A142, p. 621, 12 July 1933

Wilson, S., "The development of Poiseuille flow", Journal of Fluid Mechanics, v. 38, part 4, p. 793, 21 January 1969

Zahn, J. P. and others, "Nonlinear cellular motions in Poiseuille channel flow", Journal of Fluid Mechanics, v. 64, part 2, p. 319, 18 April 1973

INITIAL DISTRIBUTION LIST

		No. Copies
1.	Defense Documentation Center Cameron Station Alexandria, Virginia 22314	2
2.	Library, Code 0212 Naval Postgraduate School Monterey, California 93940	2
3.	Professor R. W. Bell, Code 57Be Department of Aeronautics Naval Postgraduate School Monterey, California 93940	2
4.	Professor T. H. Gawain, Code 57Gn Department of Aeronautics Naval Postgraduate School Monterey, California 93940	5
5.	Assoc. Professor M. F. Platzer, Code 57Pl Department of Aeronautics Naval Postgraduate School Monterey, California 93940	1
6.	LT William F. Harrison, USN 94 Bethpage Road Hicksville, New York 11801	2

161495
Thesis
H29344 Harrison
c.1 On the stability of
Poiseuille flow.